

Institut für Informatik und Praktische Mathematik

## **Determinisierung von $\omega$ -Automaten**

Diplomarbeit  
vorgelegt von  
Detlef Kähler

Juni 2001

Prof. Dr. Thomas Wilke

Christian-Albrechts-Universität zu Kiel

# Inhaltsverzeichnis

<b>Einleitung</b>	<b>1</b>
<b>1 Einführung</b>	<b>6</b>
1.1 Notation . . . . .	6
1.2 $\omega$ -Automaten . . . . .	6
<b>2 Berechnungsbäume</b>	<b>9</b>
2.1 $K$ -Bäume . . . . .	11
2.2 Standardberechnungsbaum . . . . .	12
2.3 01-Berechnungsbaum . . . . .	13
<b>3 Reduzierte Berechnungsbäume</b>	<b>15</b>
3.1 Linksreduktion von 01-Berechnungsbäumen . . . . .	15
3.2 Reduzierter Standardberechnungsbaum . . . . .	19
<b>4 Baumuntersuchungsverfahren</b>	<b>21</b>
4.1 Datenstrukturbäume . . . . .	22
4.2 Pfadsegmentierung . . . . .	24
4.3 Pfadverfolgung . . . . .	31
4.4 Pfadverfolgung für $\{0, 1\}$ -Bäume . . . . .	37
<b>5 Determinisierungsverfahren</b>	<b>42</b>
5.1 Verfahren nach Muller und Schupp . . . . .	43
5.2 Verfahren nach Emerson und Sistla . . . . .	46
5.3 Verfahren durch vereinfachte Pfadverfolgung . . . . .	46
<b>Fazit</b>	<b>49</b>
<b>Literatur</b>	<b>50</b>

# Einleitung

Endliche Automaten auf unendlichen Wörtern (sog.  $\omega$ -Automaten) bilden eine Erweiterung der endlichen Automaten auf endlichen Wörtern (sog.  $*$ -Automaten) [HU79]. Ein  $*$ -Automat liest ein endliches Eingabewort und akzeptiert das eingelesene Wort oder er weist es zurück. Ob der Automat das Eingabewort akzeptiert oder nicht, hängt von dem Zustand ab, in dem er sich nach dem Einlesen des Wortes befindet. Da bei einem  $\omega$ -Automaten die Eingabewörter unendlich sind, kann man nicht von einem letzten Zustand sprechen, der nach dem Einlesen eingenommen wurde. Es muß aber bei der Verarbeitung eines unendlichen Eingabewortes durch einen  $\omega$ -Automaten eine Menge von Zuständen geben, die unendlich oft angenommen werden, denn der Automat besitzt nur endlich viele Zustände. Mit Hilfe dieser Menge von unendlich oft angenommenen Zuständen kann bei  $\omega$ -Automaten das Akzeptieren von Wörtern definiert werden.

Ist  $\mathcal{A}$  ein  $\omega$ -Automat und  $\alpha$  ein Eingabewort, so beginnt  $\mathcal{A}$  in einem ausgezeichneten Anfangszustand  $q_I$  und liest den ersten Buchstaben von  $\alpha$  ein. In Abhängigkeit vom eingelesenen Buchstaben gibt es mögliche Nachfolgezustände, die  $\mathcal{A}$  einnehmen kann. Der Automat setzt die Verarbeitung in einem dieser Nachfolgezustände fort und liest den nächsten Buchstaben von  $\alpha$  ein. Fährt der Automat auf diese Weise fort, entsteht eine unendliche Folge von eingenommenen Zuständen, eine solche Folge von Zuständen wird Lauf von  $\mathcal{A}$  auf  $\alpha$  genannt. Wie oben angedeutet, gibt es in einem Lauf  $\rho$  eine Menge von Zuständen, die unendlich oft in  $\rho$  auftreten. Diese Menge nennen wir Unendlichkeitsmenge von  $\rho$  und bezeichnen sie mit  $In(\rho)$ . Mit Hilfe dieser Unendlichkeitsmenge werden Läufe in akzeptierende und verwerfende Läufe eingeteilt. Die Bedingung, nach der diese Einteilung vorgenommen wird, heißt Akzeptierbedingung. Die Akzeptierbedingung kann verschiedene Formen haben.

Die erste Form von  $\omega$ -Automaten wurde von Büchi in [Büc62] eingeführt, um ein Entscheidungsverfahren für die monadische Theorie zweiter Stufe der natürlichen Zahlen mit der Nachfolgerfunktion (S1S, second order theory of one successor) zu erhalten. Büchi verwendete dabei nichtdeterministische  $\omega$ -Automaten, die ein Eingabewort  $\alpha$  akzeptieren, wenn es einen Lauf von  $\mathcal{A}$  auf  $\alpha$  gibt, der eine ausgezeichnete Menge  $F$  von Endzuständen unendlich oft besucht, d.h. wenn es ein Lauf  $\rho$  von  $\mathcal{A}$  auf  $\alpha$  gibt, der die

$$\text{Büchi-Akzeptierbedingung} \quad In(\rho) \cap F \neq \emptyset$$

erfüllt. Ein Automat mit Büchi-Akzeptierbedingung heißt Büchi-Automat. In dem Entscheidungsverfahren von Büchi werden S1S-Formeln induktiv in nichtdeterministische Büchi-Automaten übersetzt. Auf diese Weise reduzierte Büchi das untersuchte Entscheidungsproblem auf das Leerheitsproblem für nichtdeterministische Büchi-Automaten. Das Leerheitsproblem für nichtdeterministische Büchi-Automaten ist mit Hilfe einfacher Graphenalgorithmen lösbar.

Bei der Übersetzung einer S1S Formel der Form  $\neg\phi$  in einen  $\omega$ -Automaten, wäre

eine Determinisierung hilfreich. So könnte man den zu der Formel  $\phi$  gehörenden Automaten determinisieren und diesen durch Anpassung der Akzeptierbedingung komplementieren, wie es bei  $*$ -Automaten mit Hilfe der Potenzmengenkonstruktion [RS59] geschieht. Büchi zeigte die Komplementabgeschlossenheit für die durch nichtdeterministische Büchi-Automaten akzeptierten Sprachen direkt. In seinem Beweis waren tiefgehende kombinatorische Argumente notwendig.

In [Mul63] schlug Muller eine weitere Akzeptierbedingung für  $\omega$ -Automaten vor. Diese Muller-Akzeptierbedingung besteht aus einer Menge  $\mathcal{F} = \{F_1, \dots, F_n\}$  von Zustandsmengen. Ein Eingabewort  $\alpha$  wird von  $\mathcal{A}$  akzeptiert, wenn es einen Lauf  $\rho$  von  $\mathcal{A}$  auf  $\alpha$  gibt, der die

$$\text{Muller-Akzeptierbedingung} \quad \text{In}(\rho) \in \mathcal{F}$$

erfüllt. Ein  $\omega$ -Automat mit Muller-Akzeptierbedingung heißt Muller-Automat.

McNaughton gab in [McN66] ein Determinisierungsverfahren an, mit dem aus einem nichtdeterministischen Büchi-Automaten  $\mathcal{A}$  ein deterministischer Muller-Automat  $\mathcal{D}$  konstruiert werden kann, der dieselbe Sprache akzeptiert wie  $\mathcal{A}$ . Der bei dieser Konstruktion entstehende Automat  $\mathcal{D}$  kann im schlimmsten Fall doppelt exponentielle Größe im Vergleich zum Ausgangsautomaten besitzen. Die von McNaughton für den Automaten  $\mathcal{D}$  verwendete Akzeptierbedingung wurde später von Rabin in [Rab69] formalisiert, sie wird im folgenden Rabin-Bedingung genannt.

Eine Rabin-Bedingung besteht aus einer Menge  $\Omega = \{(G_1, R_1), \dots, (G_n, R_n)\}$  von Paaren von Zustandsmengen (sog. Rabin-Paare) des zugrundeliegenden  $\omega$ -Automaten  $\mathcal{A}$ . Ein Eingabewort  $\alpha$  wird von  $\mathcal{A}$  akzeptiert, wenn es einen Lauf  $\rho$  von  $\mathcal{A}$  auf  $\alpha$  gibt, der die

$$\text{Rabin-Akzeptierbedingung} \quad \text{In}(\rho) \cap G \neq \emptyset \text{ und } \text{In}(\rho) \cap R = \emptyset$$

für ein  $(G, R) \in \Omega$  erfüllt. D.h. wenn  $\rho$  unendlich oft einen Zustand aus  $G$  und nur endlich oft einen Zustand aus  $R$  besucht. Ein  $\omega$ -Automat mit Rabin-Akzeptierbedingung heißt Rabin-Automat. Die bei den in dieser Arbeit aus nichtdeterministischen Büchi-Automaten konstruierten deterministischen Automaten sind Rabin-Automaten.

Eine erste Verbesserung der Komplexität der Determinisierung von Büchi-Automaten schlugen Emerson und Sistla in [ES84] vor. Sie gaben ein Entscheidungsverfahren für die Baumlogik  $\text{CTL}^*$  an. In ihrem Verfahren wandelten sie Teilformeln einer  $\text{CTL}^*$  Formel in einen nichtdeterministischen Büchi-Automaten um und determinisieren diesen für die weitere Verwendung in einem Baumautomaten. Das angegebene Determinisierungsverfahren lieferte einen nur noch einfach exponentiell größeren Automaten im Vergleich zum Ausgangsautomaten. Jedoch war das angegebene Verfahren nicht auf beliebige Büchi-Automaten anwendbar, sondern nutzte eine wesentliche Eigenschaft der dort betrachteten Büchi-Automaten aus: Ist ein Zustand  $q$  des Automaten mit einem endlichen Wort  $\alpha$  erreichbar, so gibt es nur genau einen Berechnungsweg, der vom Anfangszustand mit  $\alpha$

zu  $q$  führt. Daß dies keine wesentliche Einschränkung darstellt, werden wir in Abschnitt (3) zeigen. Das von Emerson und Sistla vorgeschlagene Verfahren kann damit mit einer nur kleinen Änderung auf jeden nichtdeterministischen Büchi-Automaten angewendet werden.

Die im Verfahren von Emerson und Sistla erreichte Komplexität war jedoch nicht optimal. Die untere Schranke für das Determinisieren von Büchi-Automaten ergibt sich aus einer unteren Schranke für das Komplementieren von Büchi-Automaten aus [Mic88]. Der Beweis der unteren Schranke für das Determinisieren von Büchi-Automaten kann in [Tho97] und in verbesserter Form in [Löd99] nachgelesen werden.

Das erste Determinisierungsverfahren für Büchi-Automaten, das diese untere Schranke erreicht, wurde von Safra in [Saf88] angegeben. Dieses Verfahren ist bis heute das Standardverfahren für den konstruktiven Beweis für die Optimalität dieser unteren Schranke. Zu einem nichtdeterministischen Büchi-Automaten mit  $n$  Zuständen ergibt dieses Verfahren einen äquivalenten deterministischen Rabin-Automaten mit  $2^{\mathcal{O}(n \log n)}$  Zuständen und  $n$  Rabin-Paaren. Das Verfahren kann als eine Verfeinerung der Potenzmengenkonstruktion [RS59] für \*-Automaten aufgefaßt werden. Die Zustände des konstruierten deterministischen Automaten sind keine Zustandsmengen wie bei der Potenzmengenkonstruktion, sondern Bäume, deren Knoten mit Zustandsmengen beschriftet sind.

Die Potenzmengenkonstruktion für \*-Automaten kann anschaulich interpretiert werden: betrachten wir den Zustand des Potenzmengenautomaten, der nach dem Lesen eines Wortes  $\alpha$  erreicht ist, so besteht dieser aus der Menge der Zustände, die mit  $\alpha$  vom Anfangszustand aus erreichbar sind. Befindet sich in einem Zustand  $q$  des Potenzmengenautomaten ein Endzustand des Ausgangsautomaten, so ist  $q$  ein Endzustand. Eine solche direkte Interpretation ist für die Konstruktion von Safra nicht möglich. Ein intuitives Verständnis für diese Konstruktion bleibt aus.

Muller und Schupp stellen in [MS95] ein Verfahren vor, mit dem zu einem alternierenden Baumautomaten ein äquivalenter nichtdeterministischer Baumautomat konstruiert werden kann. Mit Hilfe dieser Umwandlung lassen sich einige zuvor erzielte Ergebnisse in einem einheitlichen Rahmen auf elegante Art neu beweisen. So ergibt sich als ein Spezialfall eine optimale Konstruktion zur Determinisierung von Büchi-Automaten. Diese Konstruktion hat gegenüber der Konstruktion nach Safra den Vorteil, daß sie intuitiv erfaßt werden kann. Ein wesentlicher Schritt dazu ist die Interpretation des Verfahrens als Anwendung eines Baumuntersuchungsverfahrens auf einen Berechnungsbaum.

Neben dem rein theoretischen Interesse an einer optimalen Transformation von nichtdeterministischen Büchi-Automaten in deterministische Rabin-Automaten, ist eine solche Transformation auch im Bereich der endlichen Automaten auf unendlichen Bäumen (siehe z.B. [Tho97]) von Nutzen. In Konstruktionen für diese Automaten werden häufig implizit Wortautomaten eingesetzt, die Eigenschaften von unendlichen Pfaden durch Bäume beschreiben. Es wird bei einigen dieser Konstruktionen davon ausgegangen, daß die benutzten Wortautomaten deterministisch sind. Für die Komplexität solcher Ver-

fahren ist es somit von Bedeutung, die Komplexität der Determinisierung von Wortautomaten zu kennen. Ein Beispiel für eine solche Konstruktion ist die Komplementierung von nichtdeterministischen Baumautomaten (siehe [Tho97]).

Das Verfahren von Safra [Saf88] zur Determinisierung von Büchi-Automaten ist bisher das einzige optimale Verfahren, das direkt beschrieben vorliegt. Das Verfahren entzieht sich allerdings einem direkten intuitiven Verständnis. Das Ziel dieser Arbeit ist es, zwei optimale Determinisierungsverfahren und deren Herleitung darzustellen. Dabei wird besonderer Wert auf die Interpretierbarkeit der Arbeitsweise der entstehenden deterministischen Automaten gelegt. Das erste Verfahren ist das oben erwähnte, sich aus der Transformation von alternierenden Baumautomaten in nichtdeterministische Baumautomaten [MS95] ergebende Verfahren. Das zweite ist eine Abwandlung des Verfahrens, das Emerson und Sistla in [ES84] angaben.

Im ersten Abschnitt geben wir die grundlegenden Notationen und Begriffe an. Wir führen formal den Begriff eines  $\omega$ -Automaten ein und beschreiben die später verwendeten Automatentypen.

Im zweiten Abschnitt führen wir Berechnungsbäume ein. Mit Hilfe dieser Berechnungsbäume läßt sich das Akzeptieren von Büchi-Automaten als das Vorhandensein unendlicher Pfade eines bestimmten Typs in einem solchen Baum charakterisieren.

Im dritten Abschnitt stellen wir die für das folgende grundlegende Idee der Reduktion von Berechnungsbäumen dar. Die im zweiten Abschnitt dargestellten Berechnungsbäume können eine unbeschränkte Breite und damit eine unendliche Anzahl von unendlichen Pfaden besitzen. Mit Hilfe der dargestellten Reduktionen werden die Berechnungsbäume für die im vierten Abschnitt beschriebenen Untersuchungsverfahren zugänglich.

Im vierten Abschnitt werden wir drei unterschiedliche Untersuchungsverfahren für unendliche Bäume angeben, die zusammen mit den in Abschnitt drei definierten reduzierten Berechnungsbäumen die Grundlage für Determinisierungsverfahren von Büchi-Automaten ergeben.

Im fünften Abschnitt stellen wir schließlich auf der Grundlage der zuvor eingeführten Bäume und Untersuchungsverfahren zwei konkrete Determinisierungsverfahren für Büchi-Automaten dar.

**Danksagung** Ich danke meinem Betreuer Prof. Thomas Wilke für seine Unterstützung und meinem Freund Christof Löding für seine Hinweise zur Gestaltung dieser Arbeit.

# 1 Einführung

## 1.1 Notation

Sei  $X$  eine Menge. Ist  $X$  endlich, so bezeichnen wir mit  $|X|$  die Anzahl der Elemente von  $X$ . Wir bezeichnen mit  $X^\omega$  die Menge der unendlichen Folgen ( $\omega$ -Wörter) über  $X$ , d.h. Elemente aus  $X^\omega$  betrachten wir als Funktionen von  $\mathbb{N}$  nach  $X$ . Die Menge der endlichen Wörter beliebiger Länge über  $X$  bezeichnen wir mit  $X^*$  und die Menge der nichtleeren Wörter über  $X$  mit  $X^+$ . Für ein Wort  $\alpha = \alpha(0)\alpha(1)\dots\alpha(n-1) \in X^*$  sei  $n$  die Länge von  $\alpha$  und werde mit  $|\alpha|$  bezeichnet. Für ein  $a \in \Sigma$  sei  $|\alpha|_a$  die Anzahl der Buchstaben  $a$  im Wort  $\alpha$ . Wir identifizieren Elemente aus  $X$  mit den Wörtern der Länge 1 aus  $X^*$ . Teilmengen von  $X^\omega$  heißen  $\omega$ -Sprachen über  $X$  oder einfach Sprachen über  $X$ . Für ein  $\alpha \in X^\omega \cup X^*$  und  $i, j \in \mathbb{N}$  sei  $\alpha[i, j] = \alpha(i)\dots\alpha(j)$ . Wir verwenden runde Klammern anstelle der eckigen, wenn die entsprechende Randposition ausgeschlossen werden soll. So steht  $\alpha[i, j)$  für das Infix  $\alpha(i)\dots\alpha(j-1)$  von  $\alpha$ .

Ist auf der Menge  $X$  eine Ordnung gegeben, so bezeichnen wir mit  $<^X$  die dazugehörige lexikographische Ordnung auf  $X^+ \cup X^\omega$ , wir schreiben kurz  $<$ , falls sich die Menge  $X$  und die entsprechende Ordnung aus dem Zusammenhang ergeben. Für die lexikographische Ordnung  $<^{\{0,1\}}$  mit  $0 < 1$  schreiben wir speziell  $<^{01}$ .

Für eine zweistellige Relation  $S \subseteq X \times X$  sei  $S^{-1} = \{(x, y) \mid (y, x) \in S\}$ .

Für ein  $\alpha \in X^\omega$  definieren wir die *Unendlichkeitsmenge*  $In(\alpha)$  als die Teilmenge von  $X$ , deren Elemente unendlich oft in  $\alpha$  auftreten, d.h.

$$In(\alpha) = \{x \in X \mid \forall i \in \mathbb{N} \exists j > i (\alpha(j) = x)\}.$$

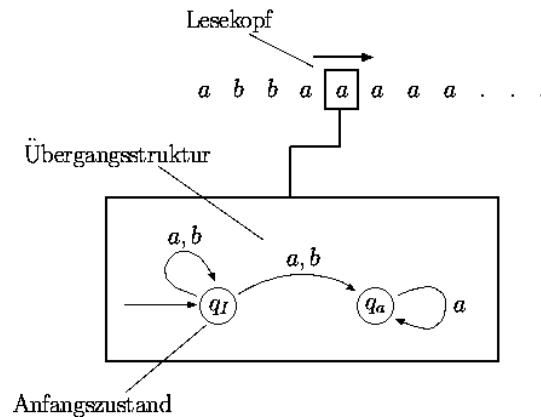
Sei  $Y$  eine weitere Menge und  $f : X \rightarrow Y$  eine Funktion von  $X$  nach  $Y$ . Für eine Teilmenge  $A$  von  $X$  setzen wir  $f(A) := \{f(x) \mid x \in A\}$ .

## 1.2 $\omega$ -Automaten

Ein  $\omega$ -Automat (kurz Automat) wird – anschaulich gesprochen – über ein Eingabewort laufen gelassen und er akzeptiert das Eingabewort oder er akzeptiert es nicht. Die Menge der akzeptierten Wörter bildet die von dem  $\omega$ -Automaten akzeptierte Sprache.

Wir veranschaulichen einen  $\omega$ -Automaten als Maschine, die einen Lesekopf besitzt und endlich viele verschiedene Zustände annehmen kann, siehe Abbildung (1). Ein  $\omega$ -Automat beginnt die Verarbeitung eines Eingabeworts in seinem Anfangszustand mit dem Lesekopf auf dem ersten Buchstaben des Eingabeworts. Er durchläuft wenn möglich die folgenden Schritte endlos:

- Einlesen des Buchstaben, auf dem der Lesekopf steht,
- Bestimmen eines Nachfolgezustands gemäß Übergangsstruktur und
- Bewegen des Lesekopfs auf den nächsten Buchstaben des Eingabeworts.

Abbildung 1: Veranschaulichung eines  $\omega$ -Automaten

Auf diese Weise entsteht bei der Verarbeitung eines Eingabeworts eine Folge von besuchten Zuständen, die Lauf genannt wird. Im dargestellten Beispiel von Abbildung (1) sind die möglichen Nachfolgezustände von  $q_I$  beim Lesen des Buchstaben  $a$  die Zustände  $q_I$  und  $q_a$ , der Automat entscheidet sich nichtdeterministisch zwischen einem dieser beiden. Ein möglicher Lauf des Automaten auf dem Eingabewort  $abbaaaa \dots$  ist etwa  $q_I q_I q_I q_I q_a q_a q_a \dots$ , ein weiterer möglicher Lauf des Automaten wäre  $q_I q_I q_I \dots$ .

Ein Lauf eines Automaten ist entweder akzeptierend oder zurückweisend. Ein Lauf ist akzeptierend, wenn er die Akzeptierbedingung des Automaten erfüllt, wir werden später auf verschiedene Arten von Akzeptierbedingungen eingehen. Ein Automat  $\mathcal{A}$  akzeptiert ein Eingabewort  $\alpha$ , wenn es einen akzeptierenden Lauf von  $\mathcal{A}$  auf  $\alpha$  gibt. Formal erhalten wir die folgende Definition eines  $\omega$ -Automaten:

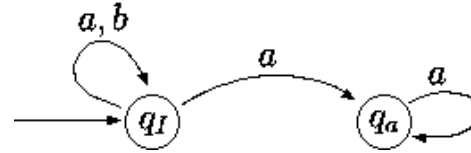
**Definition 1.1** Ein  $\omega$ -Automat (kurz Automat)  $\mathcal{A}$  hat die Form  $\mathcal{A} = (Q, \Sigma, q_I, \delta, \Omega)$ . Dabei bezeichnet  $Q$  die endliche Zustandsmenge,  $\Sigma$  das endliche Eingabealphabet,  $q_I \in Q$  den Anfangszustand,  $\delta : Q \times \Sigma \rightarrow 2^Q$  die Überföhrungsfunktion und  $\Omega$  die Akzeptierbedingung von  $\mathcal{A}$ . Auf die möglichen Formen der Akzeptierbedingung gehen wir später ein. Das 4-Tupel  $(Q, \Sigma, q_I, \delta)$  wird *Übergangsstruktur* von  $\mathcal{A}$  genannt.

Für  $q \in Q$  und  $a \in \Sigma$  nennen wir die Zustände aus  $\delta(q, a)$  auch  $a$ -Nachfolger von  $q$ . Die Überföhrungsfunktion  $\delta$  erweitern wir auf Zustandsmengen und endliche Wörter: Zu einer Teilmenge  $S \subseteq Q$ , einem Wort  $\alpha \in \Sigma^*$  und einem Buchstaben  $a \in \Sigma$  definieren wir induktiv  $\delta(S, \epsilon) := S$  und

$$\delta(S, \alpha a) := \{q \mid q \text{ ist } a\text{-Nachfolger eines } q' \in \delta(S, \alpha)\}.$$

Die Übergangsstruktur eines  $\omega$ -Automaten  $\mathcal{A}$  können wir wie oben angedeutet als Graph darstellen. Die Knotenmenge dieses Graphen entspricht der Zustandsmenge von  $\mathcal{A}$ . Aus

der Überföhrungsfunktion  $\delta$  bilden wir die Kantenmenge, d.h. eine Kante föhrt von Knoten  $q \in Q$  zum Knoten  $q' \in Q$ , wenn es einen Buchstaben  $a \in \Sigma$  gibt mit  $q' \in \delta(q, a)$ . Eine Kante  $(q, q')$  dieses Graphen beschriften wir mit der Buchstabenmenge  $\{a \in \Sigma \mid q' \in \delta(q, a)\}$ . Der Anfangszustand von  $\mathcal{A}$  wird durch einen Pfeil gekennzeichnet. Das hier dargestellte Beispiel entspricht einem  $\omega$ -Automaten mit der Übergangsstruktur  $(\{q_I, q_a\}, \{a, b\}, q_I, \delta)$ , wobei  $\delta(q_I, a) = \{q_I, q_a\}$ ,  $\delta(q_I, b) = \{q_I\}$ ,  $\delta(q_a, a) = \{q_a\}$  und  $\delta(q_a, b) = \emptyset$  ist.



**Definition 1.2** Ein *Lauf*  $\rho$  von  $\mathcal{A}$  auf  $\alpha \in \Sigma^\omega$  ist eine Zustandsfolge  $\rho \in Q^\omega$  mit  $\rho(0) = q_I$  und für  $i \geq 0$  ist  $\rho(i+1)$  ein  $\alpha(i)$ -Nachfolger von  $\rho(i)$ , d.h.  $\rho(i+1) \in \delta(\rho(i), \alpha(i))$ .

Ein *Berechnungssegment*  $\rho$  von  $\mathcal{A}$  auf dem endlichen Wort  $\alpha \in \Sigma^n$  ist eine Zustandsfolge der Länge  $n+1$  mit  $\rho(0) = q_I$  und  $\rho(i+1) \in \delta(\rho(i), \alpha(i))$  für  $0 \leq i \leq n-1$ . Ein Berechnungssegment  $\rho$  der Länge  $n$  *endet auf*  $q \in Q$ , falls  $\rho(n-1) = q$  gilt. Wir definieren für ein  $i \in \mathbb{N}$  und  $q \in Q$  die Menge

$$R_{i,q}^\alpha := \{\rho \in Q^+ \mid \rho \text{ ist Berechnungssegment von } \mathcal{A} \text{ auf } \alpha[0, i) \text{ und } \rho \text{ endet auf } q\}.$$

Ein Lauf  $\rho$  von  $\mathcal{A}$  *besucht* einen Zustand  $q \in Q$  bzw. eine Zustandsmenge  $S \subseteq Q$ , wenn für ein  $i \in \mathbb{N}$  gilt  $\rho(i) = q$  bzw.  $\rho(i) \in S$ . Ein Lauf  $\rho$  *besucht*  $q$  bzw.  $S$  *unendlich oft*, wenn  $q \in \text{In}(\rho)$  bzw.  $S \cap \text{In}(\rho) \neq \emptyset$  gilt. Ein Lauf  $\rho$  von  $\mathcal{A}$  heißt *akzeptierend*, wenn er die Akzeptierbedingung  $\Omega$  erfüllt. Der Automat  $\mathcal{A}$  *akzeptiert*  $\alpha \in \Sigma^\omega$ , wenn es einen akzeptierenden Lauf von  $\mathcal{A}$  auf  $\alpha$  gibt. Die von  $\mathcal{A}$  *akzeptierte Sprache*  $L_{\mathcal{A}}(\mathcal{A})$  besteht aus den Wörtern  $\alpha \in \Sigma^\omega$ , die von  $\mathcal{A}$  akzeptiert werden, d.h.

$$L(\mathcal{A}) = \{\alpha \in \Sigma^\omega \mid \text{es gibt einen akzeptierenden Lauf } \rho \text{ von } \mathcal{A} \text{ auf } \alpha\}.$$

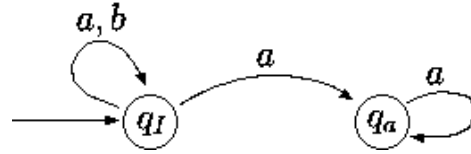
Für einen Zustand  $q \in Q$  definieren wir die *von  $q$  aus akzeptierte Sprache*  $L(q)$  als die Sprache, die von dem Automaten  $\mathcal{A}' = (Q, \Sigma, q, \delta, \Omega)$  akzeptiert wird.

Der Automat  $\mathcal{A}$  heißt *deterministisch*, wenn für alle  $q \in Q$  und  $a \in \Sigma$  gilt:  $|\delta(q, a)| = 1$ . In diesem Fall betrachten wir  $\delta$  als Funktion von  $Q \times \Sigma$  nach  $Q$ . Bei einem deterministischen Automaten  $\mathcal{A}$  gibt es offensichtlich zu jedem  $\alpha \in \Sigma^\omega$  genau einen Lauf von  $\mathcal{A}$  auf  $\alpha$ .

Die Akzeptierbedingung von  $\omega$ -Automaten kann verschiedene Formen haben. Eine *Büchi-Akzeptierbedingung*  $\Omega$  (kurz Büchi-Bedingung) besteht aus einer Menge  $F$  von Zuständen von  $\mathcal{A}$ . Die Zustände aus  $F$  werden *Endzustände* genannt. Ein Lauf  $\rho$  erfüllt die Büchi-Bedingung  $F \subseteq Q$ , wenn  $\rho$  die Menge  $F$  unendlich oft besucht. Ein  $\omega$ -Automat mit einer Büchi-Akzeptierbedingung bezeichnen wir kurz als *Büchi-Automat*.

Wir untersuchen die Determinisierung von Büchi-Automaten. D.h. wir untersuchen Verfahren, die zu einem gegebenen nichtdeterministischen Büchi-Automaten  $\mathcal{A}$  einen deterministischen Automaten  $\mathcal{B}$  konstruieren, der dieselbe Sprache akzeptiert wie  $\mathcal{A}$ .

Dabei müssen wir für den Automaten  $\mathcal{B}$  andere Akzeptierbedingungen zulassen als die Büchi-Bedingung, denn es gibt Sprachen, die von einem nichtdeterministischen Büchi-Automaten akzeptiert werden aber von keinem deterministischen Büchi-Automaten. Als Beispiel für eine solche Sprache betrachten wir  $L = \{\alpha \in \{a, b\}^\omega \mid \alpha \text{ enthält nur endlich oft } b\}$ . Der nichtdeterministische Büchi-Automat mit der rechts abgebildeten Übergangsstruktur und der Endzustandsmenge  $\{q_a\}$  akzeptiert die Sprache  $L$ .



**Bemerkung 1.1** Es gibt keinen deterministischen Büchi-Automaten, der die Sprache  $L = \{\alpha \in \{a, b\}^\omega \mid \alpha \text{ enthält nur endlich oft } b\}$  akzeptiert.

**Beweis** Sei  $\mathcal{B} = (Q, \{a, b\}, q_I, \delta, F)$  ein deterministischer Büchi-Automat, der alle Wörter aus  $L$  akzeptiert. Sei  $F'$  die Menge der erreichbaren Endzustände von  $\mathcal{B}$ , d.h.  $F'$  bestehe aus den Endzuständen  $q \in F$ , für die es ein  $\alpha \in \{a, b\}^*$  gibt mit  $q \in \delta(q_I, \alpha)$ .

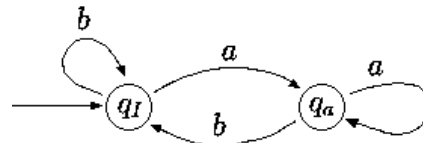
Das Wort  $baaaa\dots$  liegt offensichtlich in  $L(q)$  für jedes  $q \in F'$ , daher gibt es ein  $j_q \in \mathbb{N}$  so, daß  $\delta(q, ba^{j_q}) \in F'$  gilt. Für  $q \in F'$  setze

$$\alpha(q) := ba^{j_q}.$$

Sei  $\alpha \in \{a, b\}^*$  mit  $\delta(q_I, \alpha) \in F'$  und setze  $q_0 := \delta(q_I, \alpha)$ . Für  $i \geq 0$  setze weiterhin  $q_{i+1} := \delta(q_i, \alpha(q_i))$ .

Das Wort  $\hat{\alpha} = \alpha\alpha(q_0)\alpha(q_1)\dots$  enthält offensichtlich unendlich oft  $b$  und der Lauf von  $\mathcal{B}$  auf  $\hat{\alpha}$  besucht unendlich oft  $F'$ , also ist  $\hat{\alpha} \in L(\mathcal{B}) \setminus L$ . Ein deterministischer Büchi-Automat, der alle Wörter aus  $L$  akzeptiert, akzeptiert somit auch ein Wort, das nicht in  $L$  liegt. Es gibt also keinen deterministischen Büchi-Automaten, der  $L$  akzeptiert.  $\square$

Wir müssen also weitere Akzeptierbedingungen zulassen. Die hier untersuchten Determinisierungsverfahren verwenden für die konstruierten deterministischen Automaten eine *Rabin-Akzeptierbedingung*. Eine Rabin-Akzeptierbedingung  $\Omega$  (kurz Rabin-Bedingung) besteht aus einer Menge  $\{(G_1, R_1), \dots, (G_n, R_n)\}$  von Paaren mit  $G_i, R_i \subseteq Q$  für  $1 \leq i \leq n$ . Ein Lauf  $\rho$  erfüllt die Rabin-Bedingung  $\Omega$ , wenn es ein Paar  $(G, R) \in \Omega$  gibt mit:  $In(\rho) \cap G \neq \emptyset$  und  $In(\rho) \cap R = \emptyset$ , d.h. wenn  $\rho$  die Menge  $G$  unendlich oft und die Menge  $R$  nur endlich oft besucht. Einen  $\omega$ -Automaten mit Rabin-



Akzeptierbedingung nennen wir kurz *Rabin-Automat*. Der deterministische Rabin-Automat mit der rechts dargestellten Übergangsstruktur und der Rabin-Bedingung  $\Omega = \{(\{q_a\}, \{q_I\})\}$  akzeptiert die Sprache  $L$ .

## 2 Berechnungsbäume

Sei  $\mathcal{A}_1 = (Q, \Sigma, q_I, \delta, F)$  ein Büchi-Automat mit der in Abbildung (1) dargestell-

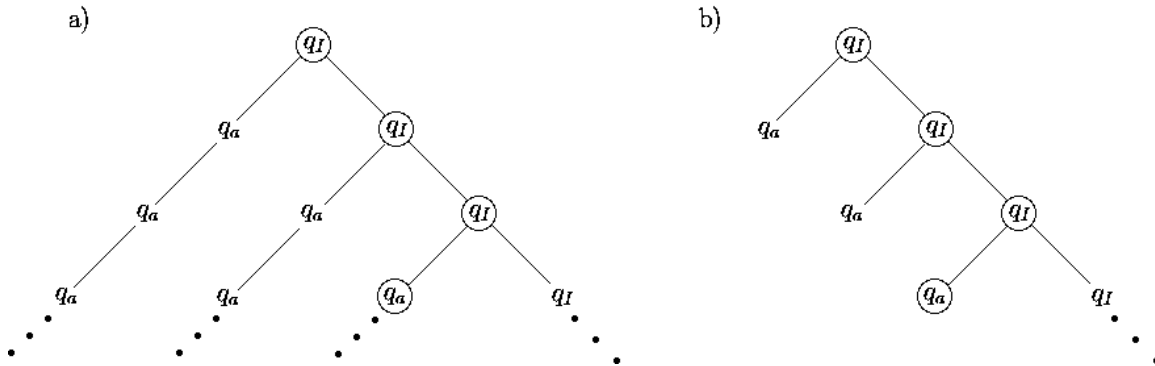
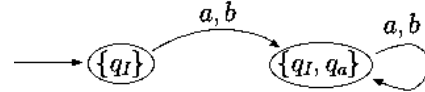


Abbildung 2: Darstellung der ersten Berechnungsschritte von  $\mathcal{A}_1$  auf den Wörtern  $\alpha = aa\dots$  und  $\beta = bb\dots$  als Baum

ten Übergangsstruktur und der Endzustandsmenge  $F = \{q_a\}$ . Es gilt  $L(\mathcal{A}_1) = \{\alpha \in \Sigma^\omega \mid \alpha \text{ enthält nur endlich oft } b\}$ . Wenden wir die von Automaten auf endlichen Wörtern bekannte Potenzmengenkonstruktion (siehe [HU79]) auf  $\mathcal{A}_1$  an, um die Übergangsstruktur eines äquivalenten deterministischen Automaten zu erhalten, ergibt sich die rechts abgebildete Struktur. Für einen  $\omega$ -Automat  $\mathcal{B}$  mit dieser Übergangsstruktur gibt es genau einen Lauf, nämlich  $\{q_I\} \{q_I, q_a\} \{q_I, q_a\} \{q_I, q_a\} \dots$ . Deshalb gilt unabhängig von der Form der Akzeptierbedingung von  $\mathcal{B}$  entweder  $L(\mathcal{B}) = \Sigma^\omega$  oder  $L(\mathcal{B}) = \emptyset$ . Bei der Potenzmengenkonstruktion geht somit Information verloren, die wir bei der Determinisierung von Büchi-Automaten benötigen.



Wir betrachten die ersten Berechnungsschritte von  $\mathcal{A}_1$  auf den Wörtern  $\alpha = aaa\dots$  und  $\beta = bbb\dots$ , um dies zu untersuchen. Hierzu stellen wir die möglichen Berechnungen, die sich zu den Wörtern  $\alpha$  und  $\beta$  ergeben, als Baum dar (siehe Abbildung (2)). Führen wir diese Bäume in der dargestellten Weise fort, so stehen deren unendliche Pfade für mögliche Läufe des Beispielautomaten auf  $\alpha$  und  $\beta$ . In dem zu  $\alpha$  gehörenden Baum erkennen wir, daß es unendliche Pfade gibt, die akzeptierende Läufe von  $\mathcal{A}_1$  auf  $\alpha$  repräsentieren, z.B. der Pfad ganz links außen. Im Baum zu  $\beta$  gibt es offensichtlich nur einen unendlichen Pfad und der steht für den zurückweisenden Lauf  $q_I q_I q_I \dots$  von  $\mathcal{A}_1$  auf  $\beta$ . Bei der Potenzmengenkonstruktion würden die Zustände, die in einer Ebene der beschriebenen Bäume auftreten, zu einer Menge zusammengefaßt. Es werden keine Pfade innerhalb der Bäume unterschieden sondern nur die mit dem eingelesenen Präfix erreichbaren Zustände betrachtet. Wir führen nun Baumarten ein mit deren Hilfe wir das Akzeptieren von Büchi-Automaten als Vorhandensein von unendlichen Pfaden bestimmter Gestalt beschreiben können. Bei den später beschriebenen Determinisierungsverfahren entstehen deterministische Automaten, die – anschaulich gesprochen –

in Berechnungsbäumen nach solchen unendlichen Pfaden suchen. Die angesprochenen Bäume stellen wir als  $K$ -Bäume dar, die wir nun einführen.

## 2.1 $K$ -Bäume

Im folgenden betrachten wir zu einer endlichen Grundmenge  $K$  Teilmengen von  $K^+$  mit deren Hilfe wir Baumstrukturen darstellen. Wir nennen eine Teilmenge  $B$  von  $K^+$  *präfixabgeschlossen*, wenn  $B$  mit jedem Wort  $\kappa \in K^+$  auch jedes nichtleere Präfix von  $\kappa$  enthält.

**Definition 2.1** Sei  $K$  eine endliche Menge. Eine präfixabgeschlossene Teilmenge  $B$  von  $K^+$ , die genau ein Element der Grundmenge  $K$  enthält, heißt  $K$ -Baum.

Die Elemente eines  $K$ -Baums  $B$  heißen *Knoten*. Das Element  $r$  der Grundmenge  $K$  in  $B$  heißt *Wurzel* von  $B$ . Ein Knoten der Form  $vk$  mit  $vk \in B$  und  $k \in K$  heißt  $k$ -Knoten. Der Knoten  $vk$  heißt *Nachfolger* von  $v$  und  $v$  heißt *Vorgänger* von  $vk$ . Allgemein heißt  $w$  *Nachfahre* von  $v$ , wenn  $v$  ein echtes Präfix von  $w$  ist. Für einen Knoten  $v \in B$  setzen wir  $\text{tiefe}(v) := |v| - 1$ . Ist der Baum  $B$  endlich und ist  $t \in \mathbb{N}$  maximal mit der Eigenschaft: es gibt einen Knoten  $v \in B$  mit  $\text{tiefe}(v) = t$ , so nennen wir  $t$  die *Tiefe* von  $B$ . Ein *endlicher Pfad* in  $B$  ist eine Knotenfolge  $\pi(0)\pi(1)\dots\pi(n)$  mit  $\pi(0)$  ist die Wurzel von  $B$  und für  $0 \leq i \leq n-1$  ist  $\pi(i+1)$  ein Nachfolger von  $\pi(i)$ . Gilt  $\pi(0) \in K^+$ , so nennen wir die Knotenfolge  $\pi(0)\pi(1)\dots\pi(n)$  ein *Pfadsegment*. Für ein Pfadsegment  $\pi$  und ein  $k \in K$  bezeichnen wir mit  $|\pi|_k$  die Anzahl der  $k$ -Knoten in  $\pi$ . Ein *unendlicher Pfad* in  $B$  ist eine unendliche Knotenfolge  $\pi(0)\pi(1)\dots$  mit  $\pi(0)$  ist die Wurzel von  $B$  und für  $i \geq 0$  ist  $\pi(i+1)$  ein Nachfolger von  $\pi(i)$ . Für zwei Knoten  $v, w \in B$  mit  $w$  ist Nachfahre von  $v$  bezeichnen wir mit  $[v, w]$  das Pfadsegment von  $v$  nach  $w$  in  $B$ . Ist  $\pi$  ein Pfad in einem  $K$ -Baum, so ist für jedes  $i < |\pi|$ :  $\pi(i+1) = \pi(i)\kappa_{i+1}$  für ein geeignetes  $\kappa_{i+1} \in K$ . Die Folge  $\sigma_\pi = \pi(0)\kappa(1)\kappa(2)\kappa(3)\dots$  nennen wir *Signatur* von  $\pi$ , d.h. die Signatur eines Pfades besteht aus den jeweils letzten Komponenten der Knoten des Pfades. Ist  $B$  ein  $K$ -Baum, so setzen wir für ein  $i \in \mathbb{N}$ :  $B_i := \{v \in B \mid |v| \leq i+1\}$ . Die Menge  $B_i$  ist offensichtlich ein  $K$ -Baum.

Mit den Knoten eines  $K$ -Baums können wir Informationen verbinden. Diese Informationen stellen wir als Beschriftung dar, die jedem Knoten aus  $B$  die zu ihm gehörende Information zuordnet. Die möglichen Beschriftungen der Knoten werden einer Beschriftungsmenge  $X$  entnommen und die Beschriftung als Abbildung von  $B$  nach  $X$  dargestellt. Wir verwenden denselben Namen für die Knotenmenge und die Beschriftungsfunktion, um überflüssige Notation zu vermeiden. Ist also  $B$  ein  $K$ -Baum und  $X$  eine gegebene Beschriftungsmenge, so bezeichnen wir für ein  $v \in B$  mit  $B(v)$  die Beschriftung des Knotens  $v$ . Ein  $K$ -Baum, dessen Knoten mit Elementen der Menge  $X$  beschriftet sind, heißt  *$X$ -beschrifteter  $K$ -Baum*.

Ist ein Pfad  $\pi$  eines  $X$ -beschrifteten  $K$ -Baums  $B$  gegeben, so heißt die Folge der Beschriftungen  $\lambda_\pi = B(\pi(0))B(\pi(1))\dots$  die *Beschriftung* von  $\pi$ .

Wir führen nun die erste Form eines Berechnungsbaums ein. Sie entspricht den oben beschriebenen Beispielbäumen. Diese Bäume bestehen aus den Berechnungssegmenten des gegebenen Automaten auf dem betrachteten Eingabewort.

## 2.2 Standardberechnungsbaum

**Definition 2.2** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, \Omega)$  ein  $\omega$ -Automat und  $\alpha \in \Sigma^\omega$ . Die Menge  $B_{\mathcal{A}, \alpha} := \{q_0 q_1 \dots q_n \in Q^+ \mid q_0 = q_I \text{ und } q_{i+1} \in \delta(q_i, \alpha(i)) \text{ für } 0 \leq i \leq n-1\}$  heißt der *Berechnungsbaum* von  $\mathcal{A}$  auf  $\alpha$

Die Menge  $B_{\mathcal{A}, \alpha}$  enthält als einziges Element aus  $Q$  den Zustand  $q_I$ . Für ein  $q_0 q_1 \dots q_n \in B_{\mathcal{A}, \alpha}$  mit  $n \geq 1$  ist offensichtlich auch  $q_0 q_1 \dots q_{n-1}$  in  $B_{\mathcal{A}, \alpha}$ . Wir erhalten damit die folgende Bemerkung.

**Bemerkung 2.1** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, \Omega)$  ein  $\omega$ -Automat und  $\alpha \in \Sigma^\omega$ . Die Menge  $B_{\mathcal{A}, \alpha}$  ist ein  $Q$ -Baum.

Wir werden im folgenden von Pfaden von Berechnungsbäumen auf Läufe oder Berechnungssegmente des zugrundeliegenden  $\omega$ -Automaten schließen. Hierbei dient uns das folgende Lemma als Hilfsmittel.

**Lemma 2.1** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, \Omega)$  ein  $\omega$ -Automat.

- i) Sei  $\alpha \in \Sigma^n$ . Seien  $S_0, S_1, \dots, S_n$  nichtleere Teilmengen von  $Q$  mit  $S_0 = \{q_I\}$ . Für  $1 \leq i \leq n$  sei jeder Zustand aus  $S_i$  ein  $\alpha(i-1)$ -Nachfolger eines Zustands aus  $S_{i-1}$ . Dann gibt es ein Berechnungssegment  $\rho$  von  $\mathcal{A}$  auf  $\alpha$  mit  $\rho(i) \in S_i$  für  $0 \leq i \leq n$ .
- ii) Sei  $\alpha \in \Sigma^\omega$ . Seien  $S_0, S_1, \dots$  nichtleere Teilmengen von  $Q$  mit  $S_0 = \{q_I\}$ . Für  $i \geq 1$  sei jeder Zustand aus  $S_i$  ein  $\alpha(i-1)$ -Nachfolger eines Zustands aus  $S_{i-1}$ . Dann gibt es einen Lauf  $\rho$  von  $\mathcal{A}$  auf  $\alpha$  mit  $\rho(i) \in S_i$  für alle  $i \in \mathbb{N}$ .

**Beweis** i): Für  $n = 0$  wähle  $\rho = q_I \in S_0$ . Für  $n > 0$  wähle ein  $q \in S_n$  beliebig und setze  $S'_{n-1} := \{p \in S_{n-1} \mid q \in \delta(p, \alpha(n-1))\}$ . Es gilt nach Voraussetzung  $S'_{n-1} \neq \emptyset$ . Für  $0 \leq i \leq n-2$  sei  $S'_i = S_i$ . Per Induktion gibt es ein Berechnungssegment  $\rho'$  von  $\mathcal{A}$  auf  $\alpha(0) \dots \alpha(n-2)$  mit  $\rho'(i) \in S'_i$  für  $0 \leq i \leq n-1$ . Setze  $\rho := \rho'q$ , dann ist  $\rho$  ein Berechnungssegment der geforderten Form von  $\mathcal{A}$  auf  $\alpha$ .

ii) Für alle  $i \in \mathbb{N}$  setze  $V_i := \{(q, i) \mid q \in S_i\}$ . Sei  $G = (V, E)$  ein gerichteter Graph mit  $V = \bigcup_{i \in \mathbb{N}} V_i$  und  $E = \{((q, i), (q', i+1)) \in V \times V \mid q' \in \delta(q, \alpha(i))\}$ . Zu einem  $v \in V$  mit  $v = (q, i)$  heißt  $q$  die  $Q$ -Komponente von  $v$  und wird mit  $q_v$  bezeichnet.

Es gilt für alle  $i > 0$ : Jeder Knoten aus  $V_i$  ist Nachfolger eines Knotens aus  $V_{i-1}$ . Weiterhin sind alle  $V_i$  endlich. Nach Königs Lemma gibt es einen unendlichen Pfad  $\pi$  in  $G$  mit  $\pi(i) \in V_i$  für alle  $i \in \mathbb{N}$ .

Sei  $\rho$  die Folge der  $Q$ -Komponenten von  $\pi$ , d.h.  $\rho(i) = q_{\pi(i)}$  für alle  $i \in \mathbb{N}$ . Es gilt offensichtlich  $\rho(i) \in S_i$  für alle  $i \in \mathbb{N}$ , es ist also insbesondere  $\rho(0) = q_I$ . Weiterhin gilt

nach Definition der Kantenmenge von  $G$ , daß  $\rho(i+1)$  ein  $\alpha(i)$ -Nachfolger von  $\rho(i)$  ist für  $i \geq 0$ . Also ist  $\rho$  ein Lauf der gewünschten Form von  $\mathcal{A}$  auf  $\alpha$ .  $\square$

Ist  $\rho$  ein Lauf von  $\mathcal{A}$  auf  $\alpha$ , so gilt offenbar für alle  $i \in \mathbb{N}$ :  $\rho(0)\rho(1)\rho(2) \dots \rho(i) \in B_{\mathcal{A},\alpha}$ , da nichtleere Präfixe von Läufen Berechnungssegmente sind. Ist andererseits  $\pi = q_0 q_0 q_1 q_0 q_1 q_2 \dots$  ein unendlicher Pfad in  $B_{\mathcal{A},\alpha}$ , so ist die Folge  $\rho = q_0 q_1 q_2 \dots$  ein Lauf von  $\mathcal{A}$  auf  $\alpha$ . Wir erhalten damit die folgende Bemerkung.

**Bemerkung 2.2** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, \Omega)$  ein  $\omega$ -Automat und  $\alpha \in \Sigma^\omega$ .

Zu einem Lauf  $\rho$  von  $\mathcal{A}$  auf  $\alpha$  gibt es einen unendlichen Pfad  $\pi$  in  $B_{\mathcal{A},\alpha}$  mit Signatur  $\rho$ , d.h.  $\sigma_\pi = \rho$ .

Zu einem unendlichen Pfad  $\pi$  in  $B_{\mathcal{A},\alpha}$  gibt es einen Lauf  $\rho$  von  $\mathcal{A}$  auf  $\alpha$ , der mit der Signatur von  $\pi$  übereinstimmt, d.h.  $\rho = \sigma_\pi$ .

Aus der obigen Bemerkung ergibt sich, daß  $\alpha$  genau dann von  $\mathcal{A}$  akzeptiert wird, wenn  $B_{\mathcal{A},\alpha}$  einen Pfad  $\pi$  enthält, dessen Signatur unendlich oft einen Endzustand enthält, d.h.  $In(\lambda_\pi) \cap F \neq \emptyset$ . Einen unendlichen Pfad  $\pi$  von  $B_{\mathcal{A},\alpha}$ , der einen akzeptierenden Lauf von  $\mathcal{A}$  auf  $\alpha$  repräsentiert, nennen wir einen *akzeptierenden Pfad* von  $B_{\mathcal{A},\alpha}$ .

## 2.3 01-Berechnungsbaum

Ob ein Lauf  $\rho$  von  $\mathcal{A}$  auf  $\alpha$  akzeptierend ist oder nicht, hängt nur davon ab, ob  $\rho$  die Menge  $F$  unendlich oft besucht. Dies führt zur Definition der *F-Projektion* eines Laufes. Dazu setzen wir für ein  $q \in Q$

$$\langle q \rangle := \begin{cases} 0 & \text{falls } q \in F \\ 1 & \text{sonst.} \end{cases}$$

Die *F-Projektion*  $\langle \rho \rangle$  eines Laufes  $\rho$  von  $\mathcal{A}$  definieren wir als  $\langle \rho \rangle := \langle \rho(0) \rangle \langle \rho(1) \rangle \dots$ . Ein Lauf  $\rho$  von  $\mathcal{A}$  ist offensichtlich genau dann akzeptierend, wenn die *F-Projektion* von  $\rho$  unendlich oft die 0 enthält, d.h.  $0 \in In(\langle \rho \rangle)$ . Für eine endliche Folge  $\rho$  von Zuständen von  $\mathcal{A}$  mit  $\rho = q_0 q_1 \dots q_n$  definieren wir die *F-Projektion*  $\langle \rho \rangle$  von  $\rho$  als  $\langle \rho \rangle := \langle q_0 \rangle \langle q_1 \rangle \dots \langle q_n \rangle$ . Wir nennen zwei Zustandsfolgen  $\rho, \rho'$  bzgl. *F-Projektion gleich* (in Zeichen  $\rho =_F \rho'$ ), wenn  $\langle \rho \rangle = \langle \rho' \rangle$  gilt.

Ebenso wie die Berechnungssegmente eines Automaten  $\mathcal{A}$  zu einem Wort  $\alpha$  einen Baum bilden gilt dies für die *F-Projektionen* der Berechnungssegmente.

**Definition 2.3** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein Büchi-Automat und  $\alpha \in \Sigma^\omega$ . Die Menge  $B_{\mathcal{A},\alpha}^{01} = \{\epsilon_0 \epsilon_1 \dots \epsilon_n \in \{0, 1\}^+ \mid \text{Es gibt ein } q_0 q_1 \dots q_n \in B_{\mathcal{A},\alpha} \text{ mit } \langle q_0 q_1 \dots q_n \rangle = \epsilon_0 \epsilon_1 \dots \epsilon_n\}$  heißt der *01-Berechnungsbaum* von  $\mathcal{A}$  auf  $\alpha$ .

**Bemerkung 2.3** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein Büchi-Automat und  $\alpha \in \Sigma^\omega$ . Die Menge  $B_{\mathcal{A},\alpha}^{01}$  ist ein  $\{0, 1\}$ -Baum.



Aus Bemerkung (2.4) ergibt sich, daß ein Automat  $\mathcal{A}$  genau dann das Wort  $\alpha$  akzeptiert, wenn  $B_{\mathcal{A},\alpha}^{01}$  einen Pfad enthält, der unendlich oft nach links verzweigt. Einen Pfad von  $B_{\mathcal{A},\alpha}^{01}$ , der unendlich oft nach links verzweigt nennen wir einen *akzeptierenden Pfad*.

Für einen Büchi-Automaten  $\mathcal{A}$  und ein Eingabewort  $\alpha$  können wir mit Hilfe der eingeführten Berechnungsbäume  $B_{\mathcal{A},\alpha}$  und  $B_{\mathcal{A},\alpha}^{01}$  entscheiden, ob  $\alpha \in L(\mathcal{A})$  gilt oder nicht.

Die Potenzmengenkonstruktion für Automaten auf endlichen Wörtern können wir so interpretieren, daß der konstruierte deterministische Automat den Berechnungsbaum zum Eingabewort ebenenweise bestimmt und die benötigten Informationen in seinen Zuständen speichert.

Es liegt also nahe zu versuchen, die Berechnungsbäume  $B_{\mathcal{A},\alpha}$  oder  $B_{\mathcal{A},\alpha}^{01}$  auf ähnliche Art zu untersuchen und die erhaltene Information in den Zuständen eines deterministischen Automaten zu speichern. Eine mögliche Strategie wäre es, Informationen zu einzelnen Pfaden der Berechnungsbäume zu speichern und mit Hilfe dieser Informationen zu überprüfen, ob diese Pfade akzeptierend sind oder nicht. Könnten wir Informationen zu allen unendlichen Pfaden durch einen deterministischen Automaten speichern, so wäre dies die Grundlage für ein Determinisierungsverfahren.

Ein Berechnungsbaum kann jedoch unendlich viele unendliche Pfade besitzen und Informationen zu allen diesen Pfaden mit Hilfe eines endlichen deterministischen Automaten zu speichern ist unmöglich. Dieses Problem lösen wir dadurch, daß wir die Berechnungsbäume so abwandeln, daß sie nur noch eine beschränkte Breite haben und damit auch nur eine beschränkte Anzahl von unendlichen Pfaden. Dies führt uns zu reduzierten Berechnungsbäumen.

### 3 Reduzierte Berechnungsbäume

Wir stellen in den beiden folgenden Abschnitten Reduktionsverfahren dar, die aus den zuvor eingeführten Berechnungsbäumen äquivalente Bäume ergeben, die nur noch eine beschränkte Breite haben. Dies gibt uns die Möglichkeit, die in Abschnitt (4) beschriebenen Baumuntersuchungsverfahren anzuwenden und so Determinisierungsverfahren für Büchi-Automaten zu erhalten.

#### 3.1 Linksreduktion von 01-Berechnungsbäumen

In einer Ebene eines 01-Berechnungsbaums  $B_{\mathcal{A},\alpha}^{01}$  kann es verschiedene Knoten  $v$  und  $v'$  geben, für deren Beschriftung  $B_{\mathcal{A},\alpha}^{01}(v) \cap B_{\mathcal{A},\alpha}^{01}(v') \neq \emptyset$  gilt, ein Zustand kann also mehrere Vorkommen in einer Ebene des Baums  $B_{\mathcal{A},\alpha}^{01}$  haben.

Eine einfache Möglichkeit der Reduktion von  $B_{\mathcal{A},\alpha}^{01}$  wäre, zu jedem Zustand  $q$ , der in einer Ebene auftritt, nur ein Vorkommen von  $q$  beizubehalten. Wir müssen dazu eine Regel angeben, nach der wir dieses Vorkommen auswählen. Sind zwei verschiedene

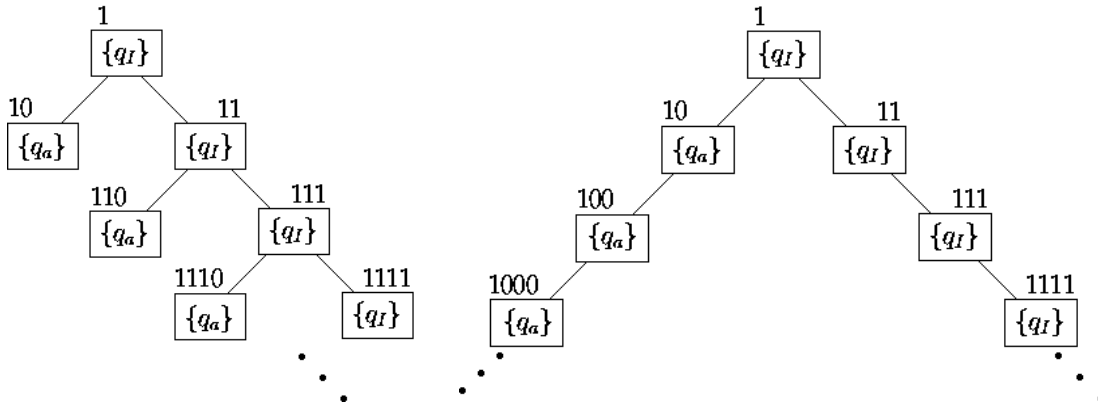


Abbildung 4: Verschiedene Arten der Vorkommenauswahl

Knoten  $v$  und  $v'$  aus einer Ebene eines  $\{0, 1\}$ -Baums gegeben, so gibt es eine natürliche Bedeutung der Aussage „ $v$  liegt links von  $v'$ “. Eine naheliegende Regel ein bestimmtes Vorkommen eines Zustands  $q$  auszuwählen ist, das jeweils rechteste oder linkeste Vorkommen von  $q$  auszuwählen. Wir betrachten dazu den Beispielautomaten  $\mathcal{A}_1$  des vorherigen Abschnitts.

Ließen wir nur das jeweils rechteste Vorkommen eines Zustands in einer Ebene, so ergäbe sich der in Abbildung (4a) abgebildete Baum zu  $\alpha = aaa \dots$ . Dieser Baum enthält offensichtlich keinen akzeptierenden Pfad, obwohl  $\alpha \in L(\mathcal{A}_1)$  gilt. Das rechteste Vorkommen eines Zustands beizubehalten führt also nicht zu einem zu  $B_{\mathcal{A}_1, \alpha}^{01}$  äquivalenten Baum.

Wählen wir stattdessen das linkeste Vorkommen eines Zustands in einer Ebene aus, so ergibt sich der in Abbildung (4b) abgebildete Baum. Dieser Baum enthält einen akzeptierenden Pfad. Betrachten wir andererseits zu einem  $\beta \in L(\mathcal{A}_1)^c$  den auf diese Weise reduzierten Baum  $B_{\mathcal{A}_1, \beta}^{01}$ , so kann dieser als Teilbaum von  $B_{\mathcal{A}_1, \beta}^{01}$  keinen akzeptierenden Pfad enthalten. Diese Linksreduktion des 01-Berechnungsbaums werden wir nun formal beschreiben und deren Korrektheit nachweisen, daß also  $B_{\mathcal{A}, \alpha}^{01}$  genau dann einen akzeptierenden Pfad enthält, wenn der reduzierte 01-Berechnungsbaum einen solchen besitzt.

Sind zwei Berechnungssegmente  $\rho$  und  $\rho'$  von  $\mathcal{A}$  auf  $\alpha[0, i)$  gegeben, so heißt  $\rho$  bzgl.  $F$ -Projektion kleiner als  $\rho'$  (in Zeichen  $\rho <_F \rho'$ ), falls  $\langle \rho \rangle <^{01} \langle \rho' \rangle$  gilt. Entsprechend heißt ein Lauf  $\rho$  von  $\mathcal{A}$  auf  $\alpha$  bzgl.  $F$ -Projektion kleiner als  $\rho'$ , falls  $\langle \rho \rangle <^{01} \langle \rho' \rangle$  gilt.

Wir definieren ebenenweise den  $2^Q$ -beschrifteten  $\{0, 1\}$ -Baum  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$ , d.h. wir bilden Mengen  $V_0, V_1, V_2 \dots$  mit  $V_i \subseteq \{0, 1\}^{i+1}$  für  $i \in \mathbb{N}$ . Die Menge  $V_i$  ist dabei die Menge der Knoten der Ebene  $i$  des Baums  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$ . Wir definieren parallel dazu Abbildungen  $\lambda_0, \lambda_1, \lambda_2 \dots$  mit  $\lambda_i : V_i \rightarrow 2^Q$  aus denen wir die Beschriftung von  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$  erhalten.

Wir setzen  $V_0 := \{\langle q_I \rangle\}$  und  $\lambda_0(\langle q_I \rangle) := \{q_I\}$ . Der Knoten  $\langle q_I \rangle$  bildet also die Wurzel von  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$ , die mit der Menge  $\{q_I\}$  beschriftet ist. Für  $i \geq 0$  bilden wir  $V_{i+1}$  und  $\lambda_{i+1}$  auf

die folgende Weise aus  $V_i$  und  $\lambda_i$ :

$\alpha(i)$ -Nachfolger bestimmen: Wir berechnen zu jedem  $v \in V_i$  die Menge der  $\alpha(i)$ -Nachfolger der zu  $v$  gehörenden Zustände, d.h. wir setzen für jedes  $v \in V_i$

$$N(v) := \delta(\lambda_i(v), \alpha(i)).$$

Wir fassen die in der Ebene  $i + 1$  vorkommenden Zustände zusammen. Dazu setzen wir

$$N_i := \bigcup_{v \in V_i} N(v).$$

*Linkeste Vorkommen bestimmen:* Zu jedem Zustand  $q \in N_i$ , d.h. zu jedem Zustand, der in der Ebene  $i + 1$  von  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$  auftritt, ermitteln wir das linkeste Vorkommen. Dazu bestimmen wir den linkesten Knoten  $v \in V_i$  für den  $q \in N(v)$  gilt, d.h. wir setzen

$$v_i^*(q) := \min\{v \in V_i \mid q \in N(v)\},$$

wobei wir zwei Elemente  $u, v \in \{0, 1\}^+$  gemäß  $<^{01}$  vergleichen.

*Nachfolgerbeschriftungen bestimmen:* Ein Knoten  $v \in V_i$  hat genau dann einen linken Nachfolger in der Ebene  $i + 1$ , wenn es Endzustände  $q \in N_i$  gibt mit  $v_i^*(q) = v$ . Ebenso gibt es den rechten Nachfolger  $v1$  genau dann in der Ebene  $i + 1$ , wenn es Nichtendzustände  $q \in N_i$  gibt mit  $v_i^*(q) = v$ . Für jeden Knoten  $v \in V_i$  unterteilen wir also die Menge der Zustände  $q$  mit  $v_i^*(q) = v$  in Endzustände und Nichtendzustände, um die Nachfolgerknoten von  $v$  bestimmen zu können, wir setzen also

$$N^0(v) := \{q \in N_i \cap F \mid v = v_i^*(q)\}$$

und

$$N^1(v) := \{q \in N_i \cap F^c \mid v = v_i^*(q)\}.$$

*Knoten und Beschriftung der Ebene  $i + 1$  bestimmen:* Anhand der Mengen  $N^0(v)$  und  $N^1(v)$  können wir für  $v \in V_i$  entscheiden, ob  $v$  einen linken oder rechten Nachfolger besitzt und wie diese Nachfolger ggf. beschriftet sind. Der Knoten  $v$  hat den Nachfolger  $v\epsilon$  mit  $\epsilon \in \{0, 1\}$ , falls  $N^\epsilon(v) \neq \emptyset$  gilt,  $v\epsilon$  wird dann mit  $N^\epsilon(v)$  beschriftet, d.h. wir setzen

$$V_{i+1} := \{v\epsilon \mid v \in V_i, \epsilon \in \{0, 1\} \text{ und } N^\epsilon(v) \neq \emptyset\}$$

und für  $v\epsilon \in V_{i+1}$  sei

$$\lambda_{i+1}(v\epsilon) := N^\epsilon(v).$$

Insgesamt definieren wir  $\tilde{B}_{\mathcal{A}, \alpha}^{01} := \bigcup_{i \in \mathbb{N}} V_i$ . Die Menge  $\tilde{B}_{\mathcal{A}, \alpha}^{01} \subseteq \{0, 1\}^+$  enthält genau ein Element der Grundmenge  $\{0, 1\}$ , nämlich  $\langle q_I \rangle$ . Ist  $v\epsilon \in V_{i+1}$  mit  $\epsilon \in \{0, 1\}$ , so ist  $v$  ein Element von  $V_i$ , da bei der Definition von  $V_{i+1}$  nur Elemente der Form  $v'\epsilon'$  mit  $v' \in V_i$  und  $\epsilon' \in \{0, 1\}$  in  $V_{i+1}$  aufgenommen werden. Wir erhalten damit

**Bemerkung 3.1** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein Büchi-Automat und  $\alpha \in \Sigma^\omega$ . Die Menge  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$  ist ein  $\{0, 1\}$ -Baum.

Zu einem Knoten  $v \in V_i$  setzen wir  $\tilde{B}_{\mathcal{A},\alpha}^{01}(v) := \lambda_i(v)$ .

**Satz 3.1** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein Büchi-Automat und  $\alpha \in \Sigma^\omega$ . Dann sind die folgenden Aussagen gleichwertig:

- i) Es gibt einen akzeptierenden Lauf von  $\mathcal{A}$  auf  $\alpha$ .
- ii) Es gibt in  $\tilde{B}_{\mathcal{A},\alpha}^{01}$  einen Pfad, der unendlich oft nach links verzweigt.

Wir zeigen zunächst Hilfsaussagen, die wir für den Beweis von Satz (3.1) benötigen. Das folgende Lemma garantiert – anschaulich gesprochen – für einen Büchi-Automaten  $\mathcal{A}$  und ein Wort  $\alpha \in L(\mathcal{A})$  die Existenz eines linkesten akzeptierenden Pfades in  $B_{\mathcal{A},\alpha}^{01}$ .

**Lemma 3.1** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein Büchi-Automat. Sei  $\alpha \in L(\mathcal{A})$ . Dann gibt es einen bzgl.  $F$ -Projektion minimalen akzeptierenden Lauf.

**Beweis** Wir konstruieren einen Teilbaum von  $B_{\mathcal{A},\alpha}$ , in dem jeder unendliche Pfad akzeptierend ist. Dazu bilden wir Indizes  $j_0, j_1, \dots$  und Mengen  $V_0, V_1, \dots$  mit:  $V_i$  ist Teilmenge der Knoten der Ebene  $j_i$  von  $B_{\mathcal{A},\alpha}$  und bilden den kleinsten Teilbaum  $B'$  von  $B_{\mathcal{A},\alpha}$ , der alle  $V_i$  enthält. Durch die spezielle Wahl der Indizes  $j_i$  und der Mengen  $V_i$  ist ein unendlicher Pfad in  $B'$  ein minimaler akzeptierender Pfad von  $\mathcal{A}$  auf  $\alpha$ .

Setze  $V_0 := \{q_I\}$ ,  $j_0 := 0$  und  $R_0 := \{\rho \in Q^\omega \mid \rho \text{ ist akzeptierender Lauf von } \mathcal{A} \text{ auf } \alpha\}$ . Da  $\alpha \in L(\mathcal{A})$  gilt, ist  $R_0 \neq \emptyset$ . Ist eine nichtleere Menge  $R$  von akzeptierenden Läufen von  $\mathcal{A}$  gegeben, so kann zu jedem  $j \in \mathbb{N}$  ein  $j' > j$  gewählt werden derart, daß ein  $\rho \in R$  existiert mit  $\rho(j') \in F$ . Für  $i \geq 0$  können wir also  $j_{i+1} > j_i$  minimal wählen mit der Eigenschaft: Es gibt ein  $\rho \in R_i$  mit  $\langle \rho(j_{i+1}) \rangle = 0$ . Setze

$$R_{i+1} := \{\rho \in R_i \mid \langle \rho(j_{i+1}) \rangle = 0\}$$

und setze weiterhin

$$V_{i+1} := \{\sigma \mid \sigma \text{ ist Berechnungssegment auf } \alpha[0, j_{i+1}) \text{ und } \exists \rho \in Q^\omega \sigma \rho \in R_{i+1}\}.$$

Sei  $V := \bigcup_{i \in \mathbb{N}} V_i$ . Setze

$$B' := \{v \in Q^+ \mid v \text{ ist Präfix eines } v' \in V\}$$

Die Menge  $B'$  ist offensichtlich ein unendlicher  $Q$ -Baum. Mit Königs Lemma ergibt sich:  $B'$  besitzt einen unendlichen Pfad  $\pi$ .

Sei  $v \in B'$  mit  $|v| = j_i + 1$ . Dann gilt  $v \in V_i$ . Denn aus  $v \in B'$  folgt: Es gibt ein  $i' \geq i$  mit  $v$  ist Präfix eines  $v' \in V_{i'}$ . Der Knoten  $v'$  hat die Form  $v' = vx$  mit  $x \in Q^*$ . Zu  $v'$  gibt es ein  $\rho \in Q^\omega$  mit  $v'\rho \in R_{i'} \subseteq R_i$ , somit ist  $vx\rho \in R_i$  und damit  $v \in V_i$ .

Es gilt damit  $\pi(j_i) \in V_i$  für alle  $i \in \mathbb{N}$ , also ist  $\pi$  akzeptierender Pfad von  $B_{\mathcal{A},\alpha}$ .

Annahme: Es gibt einen akzeptierenden Lauf  $\rho'$  mit  $\rho' <_F \rho_\pi$ . Dann gäbe es ein minimales  $j > 0$  mit  $\langle \rho'(j) \rangle \neq \langle \rho_\pi(j) \rangle$  und  $\langle \rho'(j) \rangle = 0$ . Es gäbe dann ein  $i \in \mathbb{N}$  mit  $j_i < j < j_{i+1}$  im Widerspruch zur Wahl von  $j_{i+1}$ . Also ist  $\rho_\pi$  ein bzgl.  $F$ -Projektion minimaler akzeptierender Lauf.  $\square$

**Bemerkung 3.2** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein nichtdeterministischer Büchi-Automat. Sei  $\alpha \in L(\mathcal{A})$ . Sei  $\rho$  ein bzgl.  $F$ -Projektion minimaler akzeptierender Lauf von  $\mathcal{A}$  auf  $\alpha$ . Dann ist  $\rho(0)\rho(1)\dots\rho(i)$  für jedes  $i \in \mathbb{N}$  ein bzgl.  $F$ -Projektion minimales Element von  $R_{i,\rho(i)}^\alpha$ .

**Beweis** Gäbe es ein  $i \in \mathbb{N}$  so, daß es ein  $\rho' \in R_{i,\rho(i)}$  gibt mit  $\rho' <_F \rho(0)\dots\rho(i)$ , so wäre  $\rho'(i+1)\rho(i+2)\dots$  ein akzeptierender Lauf von  $\mathcal{A}$  auf  $\alpha$  mit kleinerer  $F$ -Projektion als  $\rho$  im Widerspruch zur Wahl von  $\rho$ .  $\square$

**Bemerkung 3.3** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein nichtdeterministischer Büchi-Automat. Sei  $\alpha \in \Sigma^\omega$ . Sei  $j \in \mathbb{N}$ ,  $q \in \delta(q_I, \alpha[0, j])$ . Sei  $\rho$  bzgl.  $F$ -Projektion minimales Element von  $R_{j,q}^\alpha$ . Dann gilt:  $\langle \rho \rangle \in \tilde{B}_{\mathcal{A},\alpha}^{01}$  und  $q \in \tilde{B}_{\mathcal{A},\alpha}^{01}(\langle \rho \rangle)$ .

**Beweis** Für  $j = 0$  ist  $q = q_I$  und  $\rho = q_I$ . Offensichtlich ist  $\langle q_I \rangle \in \tilde{B}_{\mathcal{A},\alpha}^{01}$  und  $q_I \in \tilde{B}_{\mathcal{A},\alpha}^{01}(\langle q_I \rangle)$ . Für  $j > 0$  hat  $\rho$  die Form  $\rho = \rho'q$  mit  $\rho' \in R_{j-1,q'}^\alpha$  für ein  $q' \in Q$ . Das Berechnungssegment  $\rho'$  ist bzgl.  $F$ -Projektion minimales Element von  $R_{j-1,q'}$ , da ansonsten  $\rho'q = \rho$  nicht minimal bzgl.  $F$ -Projektion in  $R_{j,q}^\alpha$  wäre. Also gilt per Induktion  $\langle \rho' \rangle \in \tilde{B}_{\mathcal{A},\alpha}^{01}$  und  $q' \in \tilde{B}_{\mathcal{A},\alpha}^{01}(\langle \rho' \rangle)$ . Bei der Bestimmung von  $V_j$  bei der Definition von  $\tilde{B}_{\mathcal{A},\alpha}^{01}$  ist  $q \in N(\langle \rho' \rangle)$ , also gilt  $v_{j-1}^*(q) \leq \langle \rho' \rangle$ . Wäre  $v_{j-1}^*(q) < \langle \rho' \rangle$ , so gäbe es ein Element  $\rho'' \in R_{j,q}^\alpha$  mit  $\langle \rho'' \rangle = v_{j-1}^*(q) < \langle \rho' \rangle$  im Widerspruch zur Minimalität von  $\rho'$ .  $\square$

**Beweis von Satz (3.1)** i)  $\Rightarrow$  ii): Es gebe einen akzeptierenden Lauf von  $\mathcal{A}$  auf  $\alpha$ . Es gibt einen bzgl.  $F$ -Projektion minimalen akzeptierenden Lauf  $\rho$  von  $\mathcal{A}$  auf  $\alpha$ , siehe Lemma (3.1). Setze  $\pi_\rho := \langle \rho(0) \rangle \langle \rho(0)\rho(1) \rangle \langle \rho(0)\rho(1)\rho(2) \rangle \dots$ . Aus den Bemerkungen (3.2) und (3.3) ergibt sich, daß  $\pi_\rho$  ein Knotenfolge in  $\tilde{B}_{\mathcal{A},\alpha}^{01}$  ist. Offensichtlich ist  $\pi_\rho$  ein Pfad in  $\tilde{B}_{\mathcal{A},\alpha}^{01}$ . Da  $\rho$  ein akzeptierender Lauf ist, gilt  $\langle \rho(i) \rangle = 0$  für unendlich viele verschiedene  $i \in \mathbb{N}$ , also zweigt  $\pi_\rho$  unendlich oft nach links ab.

$\neg$ i)  $\Rightarrow$   $\neg$ ii): Gibt es keinen akzeptierenden Lauf von  $\mathcal{A}$  auf  $\alpha$ , so gibt es nach Bemerkung (2.4) keinen Pfad in  $B_{\mathcal{A},\alpha}^{01}$ , der unendlich oft nach links verzweigt. Da  $\tilde{B}_{\mathcal{A},\alpha}^{01}$  Teilbaum von  $B_{\mathcal{A},\alpha}^{01}$  ist, gibt es auch in  $\tilde{B}_{\mathcal{A},\alpha}^{01}$  keinen unendlich oft nach links verzweigenden Pfad.  $\square$

### 3.2 Reduzierter Standardberechnungsbaum

Wir können den Standardberechnungsbaum  $B_{\mathcal{A},\alpha}$  auf dieselbe Art reduzieren, wie wir den 01-Berechnungsbaum  $B_{\mathcal{A},\alpha}^{01}$  reduziert haben. D.h. gibt es in einer Ebene von  $B_{\mathcal{A},\alpha}$  verschiedene Knoten, die auf denselben Zustand enden, so wählen wir genau einen dieser Knoten aus, den wir in dieser Ebene belassen. Treffen wir diese Auswahl in der richtigen Weise, erhalten wir einen zu  $B_{\mathcal{A},\alpha}$  äquivalenten Baum  $\tilde{B}_{\mathcal{A},\alpha}$ .

Im folgenden beschreiben wir eine mögliche Art der Reduktion, die sich an die Reduktion des 01-Berechnungsbaums  $B_{\mathcal{A},\alpha}^{01}$  anlehnt. Anschaulich gesehen, wählen wir zu jedem Knoten  $v \in \tilde{B}_{\mathcal{A},\alpha}^{01}$  der Ebene  $i$  und jedem  $q \in \tilde{B}_{\mathcal{A},\alpha}^{01}(v)$  einen Knoten  $w$  der Ebene  $i$  von  $B_{\mathcal{A},\alpha}$  aus mit  $w(i-1) = q$  und  $\langle w \rangle = v$ .

Für  $v, v' \in Q^n$  mit  $n > 0$  und  $v(n-1) \neq v'(n-1)$  definieren wir:

$$v < v' :\iff \langle v \rangle < \langle v' \rangle \text{ oder } (\langle v \rangle = \langle v' \rangle \text{ und } v(n-1) <_Q v'(n-1)),$$

wobei  $<_Q$  eine beliebige aber feste Ordnung auf  $Q$  sei.

Wir definieren ebenenweise den  $Q$ -Baum  $\tilde{B}_{\mathcal{A},\alpha}$ , d.h. wir bilden Mengen  $V_0, V_1, V_2 \dots$  mit  $V_i \subseteq Q^{i+1}$  für  $i \in \mathbb{N}$ . Die Menge  $V_i$  ist dabei die Menge der Knoten der Ebene  $i$  des Baums  $\tilde{B}_{\mathcal{A},\alpha}$ .

Wir setzen  $V_0 := \{q_I\}$ . Der Knoten  $q_I$  bildet die Wurzel von  $\tilde{B}_{\mathcal{A},\alpha}$ . Für  $i \geq 0$  bilden wir  $V_{i+1}$  auf die folgende Weise aus  $V_i$ :

$\alpha(i)$ -Nachfolger bestimmen: Wir berechnen zu jedem  $v \in V_i$  die Menge der  $\alpha(i)$ -Nachfolger von  $v(i)$ , d.h. wir setzen für jedes  $v \in V_i$

$$N(v) := \delta(v(i), \alpha(i)).$$

Wir fassen die in der Ebene  $i+1$  vorkommenden Zustände zusammen, dazu setzen wir

$$N_i := \bigcup_{v \in V_i} N(v).$$

*Minimales Vorkommen bestimmen:* Zu jedem Zustand  $q \in N_i$ , d.h. zu jedem Zustand, der in der Ebene  $i+1$  von  $\tilde{B}_{\mathcal{A},\alpha}$  auftritt, ermitteln wir ein minimales Vorkommen. Dazu bestimmen wir den bzgl.  $<$  kleinsten Knoten  $v \in V_i$  für den  $q \in N(v)$  gilt, d.h. wir setzen

$$v_i^*(q) := \min_{<} \{v \in V_i \mid q \in N(v)\}.$$

Da zwei verschiedene Knoten  $u, v$  von  $V_i$  auf unterschiedliche Zustände enden, ist  $v_i^*(q)$  wohldefiniert.

*Knoten der Ebene  $i+1$  bestimmen:* Setze

$$V_{i+1} := \{v_i^*(q)q \mid q \in N_i\}.$$

Insgesamt definieren wir  $\tilde{B}_{\mathcal{A},\alpha} := \bigcup_{i \in \mathbb{N}} V_i$ . Die Menge  $\tilde{B}_{\mathcal{A},\alpha}^{01} \subseteq Q^+$  enthält genau ein Element der Grundmenge  $Q$ , nämlich  $q_I$ . Ist  $vq \in V_{i+1}$  mit  $q \in Q$ , so ist  $v$  ein Element von  $V_i$ , da bei der Definition von  $V_{i+1}$  nur Elemente der Form  $v'q'$  mit  $v' \in V_i$  und  $q' \in Q$  in  $V_{i+1}$  aufgenommen werden. Wir erhalten damit

**Bemerkung 3.4** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein Büchi-Automat und  $\alpha \in \Sigma^\omega$ . Die Menge  $\tilde{B}_{\mathcal{A},\alpha}$  ist ein  $Q$ -Baum.

**Bemerkung 3.5** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein Büchi-Automat und  $\alpha \in \Sigma^\omega$ . Dann gilt:

- i) Für  $i \in \mathbb{N}$  und  $q \in \delta(q_I, \alpha[0, i])$  gilt:  $\langle v_i(q) \rangle = v_i^{01}(q)$ .
- ii) Für  $v \in \tilde{B}_{\mathcal{A},\alpha}^{01}$  gibt es ein  $\rho \in \tilde{B}_{\mathcal{A},\alpha}$  mit  $\langle \rho \rangle = v$ .

iii) Für einen unendlichen Pfad  $\pi$  in  $\tilde{B}_{\mathcal{A},\alpha}^{01}$  gibt es einen Pfad  $\pi'$  in  $\tilde{B}_{\mathcal{A},\alpha}$  mit  $\langle \sigma_{\pi'} \rangle = \sigma_{\pi}$ .

**Beweis i):** Sei  $i \in \mathbb{N}$  und  $q \in \delta(q_I, \alpha[0, i])$ . Ist  $i = 0$  so ist  $q = q_I$  und es gilt offensichtlich  $\langle v_0^*(q_I) \rangle = v_0^{01}(q_I)$ . Für  $i > 0$  gilt für alle  $p \in \delta(q_I, \alpha[0, i-1])$  mit  $q \in \delta(p, \alpha(i-1))$  per Induktion  $\langle v_{i-1}(p) \rangle = v_{i-1}^{01}(p)$ . Es gilt also  $p_{i-1}^*(q) = v_{i-1}^*(q)$  (siehe Definition von  $\tilde{B}_{\mathcal{A},\alpha}$  und  $\tilde{B}_{\mathcal{A},\alpha}^{01}$ ). Damit ist  $\langle v_i(q) \rangle = p_{i-1}^*(q) \langle q \rangle = v_{i-1}^*(q) \langle q \rangle = v_i^{01}(q)$ .

ii): Sei  $v \in \tilde{B}_{\mathcal{A},\alpha}^{01}$  mit  $|v| = i+1$ . Es gibt ein  $q \in \tilde{B}_{\mathcal{A},\alpha}^{01}(v)$  und mit Teil i) gilt  $\langle v_i(q) \rangle = v_i^{01} = v$ . Wähle  $\rho = v_i(q)$ .

iii): Sei  $\pi$  ein unendlicher Pfad in  $\tilde{B}_{\mathcal{A},\alpha}^{01}$ . Setze  $V_i := \{\rho \in \tilde{B}_{\mathcal{A},\alpha} \mid \langle \rho \rangle = \pi(i) \text{ für } i \in \mathbb{N}\}$ . Nach Teil ii) gilt  $V_i \neq \emptyset$  für alle  $i \in \mathbb{N}$ . Es gilt für ein  $\rho = \rho_0 \rho_1 \dots \rho_i \in V_i$ :  $\rho_0 \dots \rho_{i-1} \in V_{i-1}$ . Weiterhin gilt  $V_0 = \{q_I\}$ , also ist die Menge  $B' = \bigcup_{i \in \mathbb{N}} V_i$  ein unendlicher  $Q$ -Baum. Mit Königs Lemma erhält man einen unendlichen Pfad  $\pi'$  in  $B'$ . Es gilt offensichtlich  $\pi'(i) \in V_i$  für alle  $i \in \mathbb{N}$  und damit  $\langle \pi'(i) \rangle = \pi(i)$ . Es gilt also  $\langle \sigma_{\pi'} \rangle = \sigma_{\pi}$ .  $\square$

**Satz 3.2** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein Büchi-Automat und  $\alpha \in \Sigma^\omega$ . Dann sind die folgenden Aussagen gleichwertig:

- i) Es gibt einen akzeptierenden Lauf von  $\mathcal{A}$  auf  $\alpha$ .
- ii) Es gibt in  $\tilde{B}_{\mathcal{A},\alpha}$  einen akzeptierenden Pfad.

**Beweis i)  $\Rightarrow$  ii):** Es gebe einen akzeptierenden Lauf von  $\mathcal{A}$  auf  $\alpha$ . Nach Satz (3.1) gibt es in  $\tilde{B}_{\mathcal{A},\alpha}^{01}$  einen akzeptierenden Pfad  $\pi$ . Nach Bemerkung (3.5) gibt es in  $\tilde{B}_{\mathcal{A},\alpha}$  einen Pfad  $\pi'$  mit  $\langle \sigma_{\pi'} \rangle = \sigma_{\pi}$ , der Pfad  $\pi'$  ist damit akzeptierend.

$\neg$ i)  $\Rightarrow$   $\neg$ ii): Es gebe keinen akzeptierenden Lauf von  $\mathcal{A}$  auf  $\alpha$ . Damit gibt es nach Bemerkung (2.2) in  $B_{\mathcal{A},\alpha}$  keinen akzeptierenden Pfad. In  $\tilde{B}_{\mathcal{A},\alpha}$  kann es als Teilbaum von  $\tilde{B}_{\mathcal{A},\alpha}$  somit keinen akzeptierenden Pfad geben.  $\square$

## 4 Baumuntersuchungsverfahren

Wir stellen in diesem Abschnitt drei Untersuchungsverfahren für unendliche  $K$ -Bäume dar, die die Grundlage für die später beschriebenen Determinisierungsverfahren bilden. Bei diesen Untersuchungsverfahren wird zu der Folge  $B_0, B_1, \dots$  eines unendlichen  $K$ -Baums  $B$  eine Folge  $C_0, C_1, \dots$  von *Kodierungsbäumen* gebildet. Der Kodierungsbaum  $C_i$  spiegelt dabei die Struktur von  $B_i$  wider so, daß an Hand der Kodierungsbaumfolge zu erkennen ist, ob der untersuchte Baum  $B$  einen unendlichen Pfad einer bestimmten Struktur besitzt. Der Unterschied zwischen den Untersuchungsverfahren besteht in der Art der Kodierungsbäume und in den Voraussetzungen, die an den untersuchten Baum  $B$  gestellt werden.

## 4.1 Datenstrukturbäume

Wir geben hier die grundlegenden Definitionen und Notationen für die in den Baumuntersuchungs- und Determinisierungsverfahren verwendeten Baumstrukturen an. Dabei sei  $\mathcal{V}$  grundsätzlich eine endliche Menge von Knotennamen.

**Definition 4.1** Ein *Baum*  $B$  über  $\mathcal{V}$  ist ein Tripel  $B = (V, E, r)$  mit den folgenden Eigenschaften:

- $V \subset \mathcal{V}$ ,
- Das Paar  $(V, E)$  ist ein gerichteter Graph,
- $r \in V$ ,
- zu jedem  $v \in V$  gibt es genau einen Pfad von  $r$  nach  $v$  in  $(V, E)$ ,

Die Menge  $V$  heißt *Knotenmenge* von  $B$ , der Knoten  $r$  heißt *Wurzel* von  $B$ .

Für einen Knotennamen  $v \in \mathcal{V}$  schreiben wir  $v \in B$ , falls  $v \in V$  gilt. Ein Knotenname  $v \in \mathcal{V}$  mit  $v \notin B$  heißt *frei*. Sind  $v, w \in B$  mit  $(v, w) \in E$ , so heißt  $w$  *Nachfolger* von  $v$  und  $v$  heißt *Vorgänger* von  $w$ . Liegt ein Knoten  $v \neq w$  auf dem Pfad von der Wurzel  $r$  nach  $w$ , so nennen wir  $w$  *Nachfahre* von  $v$ . Mit  $|B|$  bezeichnen wir die Anzahl der Knoten in  $B$ .

Ein *beschrifteter Baum* ist ein Baum  $B$  zusammen mit einer Menge von  $n$  Beschriftungsfunktionen  $\{\lambda_i : \mathcal{V} \rightarrow X_i \mid X_i \text{ ist eine Menge } 1 \leq i \leq n\}$  für ein  $n \in \mathbb{N}$ . Die Funktionen  $\lambda_i$  heißen *Beschriftungskomponenten* von  $B$ .

Ein *geordneter Baum* ist ein Baum  $B$  zusammen mit einer transitiven Relation  $S \subseteq V \times V$  mit der folgenden Eigenschaft: Haben zwei Knoten  $v, w \in B$  denselben Vorgänger, so gilt entweder  $(v, w) \in S$  oder  $(w, v) \in S$ . Die Relation  $S$  nennen wir *Geschwisterordnung*.

In einem geordneten Baum  $B$  über  $\mathcal{V}$ , können wir die Geschwisterordnung  $S$  auf verschiedene Weise zu einer Ordnung auf ganz  $V$  ausdehnen. Dies beschreiben wir im folgenden. Für  $B$  gibt es verschiedene Möglichkeiten die Knoten in eine Reihenfolge zu bringen, eine ergibt sich aus der *postorder* Durchlaufstrategie.

Bei der postorder Durchlaufstrategie werden zunächst die Teilbäume der Nachfolger der Wurzel in der durch  $S$  festgelegten Reihenfolge gemäß postorder Strategie durchlaufen und daran anschließend die Wurzel besucht. Die sich daraus ergebende Ordnung bezeichnen wir mit  $<_{\text{post}}^S$ , ergibt sich  $S$  aus dem Zusammenhang, so schreiben wir  $<_{\text{post}}$ . In Abbildung (5) ist ein geordneter Baum dargestellt, die Geschwisterordnung ist durch das  $<$ -Zeichen angedeutet, die Numerierung der Knoten beschreibt die dazugehörige Ordnung  $<_{\text{post}}$ .

Bei der Abschätzung der Größe der später konstruierten Automaten benötigen wir folgende Bemerkung:

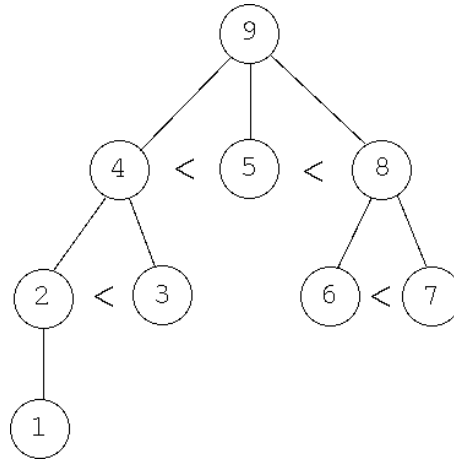


Abbildung 5: Erweiterung einer Geschwisterordnung auf alle Knoten durch postorder Durchlaufstrategie

**Bemerkung 4.1** Sei  $\mathcal{V}$  eine endliche Menge von Knotennamen. Sei  $n \leq |\mathcal{V}|$ . Sei  $\mathcal{B}$  die Menge der geordneten Bäume über  $\mathcal{V}$  mit höchstens  $n$  Knoten. Dann gilt  $|\mathcal{B}| \in 2^{\mathcal{O}(n \log n)}$ .

**Beweis** Einen Baum  $B \in \mathcal{B}$  können wir durch drei Abbildungen eindeutig darstellen. Die erste Abbildung beschreibt die Zuordnung von Knoten zu Knotennamen, sie ist eine Abbildung von  $\{1, \dots, n\}$  nach  $\mathcal{V} \cup \{\perp\}$ , wobei  $\perp$  als Wert für nicht vorhandene Knoten verwendet wird. Es gibt

$$(|\mathcal{V}| + 1)^n = 2^{n \log(|\mathcal{V}|+1)} = 2^{n(\log n + x)} \in 2^{\mathcal{O}(n \log n)}$$

solche Abbildungen.

Die Kantenmenge  $E$  von  $B$  stellen wir als Funktion von  $\{1, \dots, n\}$  in  $\{0, 1, \dots, n\}$  dar. Wobei wir einem Knoten  $i$  seinen eindeutigen Vorgänger zuordnen. Der Wert 0 wird als Wert für Knoten ohne Vorgänger verwendet. Es gibt  $(n+1)^n \in 2^{\mathcal{O}(n \log n)}$  solche Funktionen.

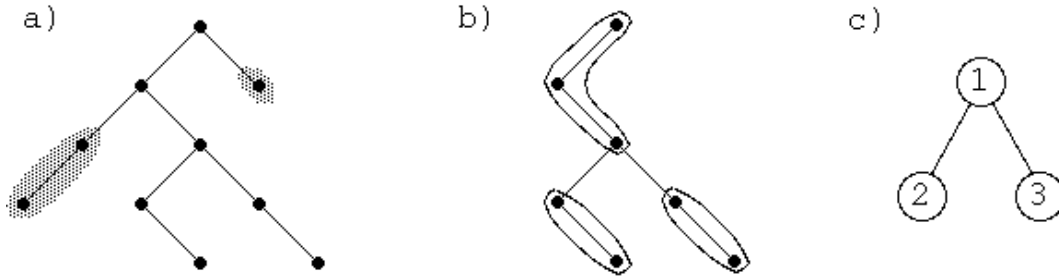
Die Geschwisterordnung stellen wir ebenfalls als Funktion von  $\{1, \dots, n\}$  in  $\{0, 1, \dots, n\}$  dar. Hier ordnen wir einem Knoten sein gemäß Geschwisterordnung nächstgrößeres Geschwister zu. Der Wert 0 dient wieder dazu, die Abbildung total zu machen. Es gibt  $(n+1)^n \in 2^{\mathcal{O}(n \log n)}$  solche Funktionen.

Insgesamt können wir also die Anzahl der Bäume aus  $\mathcal{B}$  wie folgt abschätzen:

$$|\mathcal{B}| \in 2^{\mathcal{O}(n \log n)} 2^{\mathcal{O}(n \log n)} 2^{\mathcal{O}(n \log n)} \in 2^{\mathcal{O}(n \log n)}.$$

Dies war zu zeigen.  $\square$

Für spätere Abschätzung benötigen wir die folgende Bemerkung:

Abbildung 6: Kodierung eines endlichen  $\{0,1\}$ -Baums durch Pfadsegmentierung

**Bemerkung 4.2** Sei  $B$  ein endlicher Baum über  $\mathcal{V}$ . Es gelte für alle Knoten  $v \in B$ :  $v$  hat genau zwei Nachfolger oder  $v$  ist ein Blatt. Sei  $l$  die Anzahl der Blätter von  $B$ . Dann gilt  $|B| = 2l - 1$ .

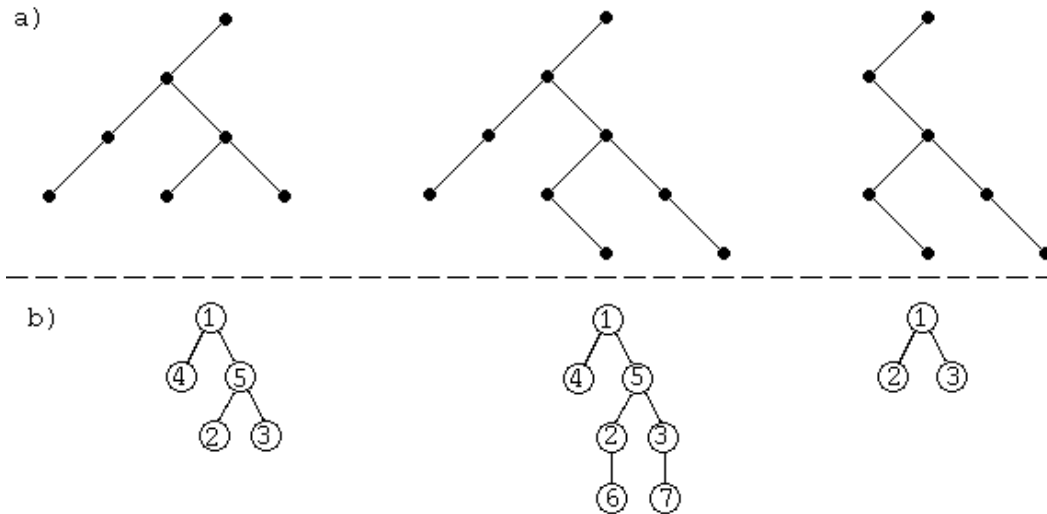
**Beweis** Ist  $l = 1$  so gilt die Behauptung offensichtlich. Ist  $l > 1$  so gibt es zwei Blätter  $v_1, v_2$  die einen gemeinsamen Vorgänger  $v$  haben. Entfernen wir die Blätter  $v_1$  und  $v_2$  aus  $B$ , so erhalten wir einen Baum  $B'$  mit  $l - 1$  Blättern und  $|B| - 2$  Knoten. Es folgt:  $|B| = |B'| + 2 = 2(l - 1) - 1 + 2 = 2l - 1$ . Dies war zu zeigen.  $\square$

## 4.2 Pfadsegmentierung

In Abbildung (6a) sei der Baum  $B_4$  eines zu untersuchenden unendlichen  $\{0, 1\}$ -Baums  $B$  dargestellt. Wir sind nur an den unendlichen Pfaden von  $B$  interessiert. Die unterlegten Knoten von  $B_4$  können auf keinem unendlichen Pfad liegen, da ihre Knoten schon keine Nachfahren in der Ebene 4 von  $B$  haben. Informationen zu diesen Knoten müssen wir im Kodierungsbaum  $S_4$  zu  $B_4$  also nicht speichern. In Abbildung (6b) ist der bereinigte Baum  $B'_4$  dargestellt, d.h. es wurden die unterlegten Teile von  $B_4$  weggelassen. Wir können für keinen Knoten von  $B'_4$  ausschließen, daß er auf einem unendlichen Pfad von  $B$  liegt. Im Kodierungsbaum  $S_4$  müssen wir den gesamten Baum  $B'_4$  berücksichtigen.

Die Kodierungsbäume müssen in der Größe beschränkt bleiben, da sie letztendlich Zustände eines endlichen Automaten sein sollen. Wir fassen deshalb Pfadsegmente in  $B'_4$ , die keine Abzweigungen besitzen, zu einem Knoten in  $S_4$  zusammen. Die Verzweigungsstruktur von  $S_4$  spiegelt die Beziehung der repräsentierten Pfadsegmente untereinander wider. In Abbildung (6b) sind die Pfadsegmente von  $B'_4$ , die keine Abzweigungen haben, umrandet. Ein sich daraus ergebender Kodierungsbaum  $S_4$  ist in Abbildung (6c) wiedergegeben.

Wir untersuchen den unendlichen Baum  $B$  ebenenweise, d.h. wir bilden zu der Folge  $B'_0, B'_1, B'_2, \dots$  der bereinigten Teilbäume die Folge  $S_0, S_1, S_2, \dots$  von dazugehörigen Kodierungsbäumen. Anhand der Folge der Kodierungsbäume entscheiden wir, ob  $B$  einen unendlich oft nach links verzweigenden Pfad besitzt.

Abbildung 7: Übergang vom Pfadsegmentierungsbaum  $S_3$  zum Baum  $S_4$ 

Um den Kodierungsbaum  $S_{i+1}$  zu bestimmen, gehen wir von  $S_i$  aus und integrieren die Informationen der Ebene  $i+1$  von  $B$ . Als Beispiel betrachten wir die Bäume  $B'_3$  und  $B'_4$  zu dem in Abbildung (6) dargestellten Baum und den Übergang zwischen diesen. Um  $B'_4$  zu erhalten, erweitern wir  $B'_3$  um die Knoten der Ebene 4 von  $B$ . Anschließend streichen wir die Knoten, die nicht auf einem Pfad von der Wurzel zu einem neuen Blatt liegen. Beim Streichen von Knoten können neue Pfadsegmente ohne Abzweigungen entstehen, siehe Abbildung (7a).

Dasselbe Vorgehen können wir auch anwenden, um aus  $S_3$  den Baum  $S_4$  zu erhalten. D.h. wir fügen zu den Blättern von  $S_3$  entsprechend der Struktur von  $B_4$  Nachfolger hinzu und entfernen diejenigen Knoten, die kein neues Blatt als Nachfahren haben. Entsteht hierbei ein Knoten  $v$ , der nur einen Nachfolger  $w$  hat, so fassen wir die Knoten  $v$  und  $w$  zusammen und  $v$  repräsentiert dann auch das Pfadsegment von  $w$ . In diesem Fall sagen wir: "Der Knoten  $v$  übernimmt den Knoten  $w$ ", siehe Abbildung (7b).

Bilden wir auf diese Weise die Folge  $S_0, S_1, \dots$  so finden wir in  $S_i$  alle Pfade der Länge  $i$  von  $B_i$  in Pfadsegmente unterteilt wieder. Insgesamt decken wir so alle unendlichen Pfade von  $B$  ab, d.h. es wird für jeden unendlichen Pfad  $\pi$  von  $B$  einen Knotennamen  $v_\pi$  geben, der immer größere Teile von  $\pi$  repräsentiert. Der Knoten  $v_\pi$  kann also von einer bestimmten Stelle an als Repräsentant von  $\pi$  in der Folge der Kodierungsbäume angesehen werden.

Um entscheiden zu können, ob  $B$  einen unendlich oft nach links verzweigenden Pfad besitzt, müssen wir in den Knoten der Kodierungsbäume zusätzliche Informationen zu der Struktur der repräsentierten Pfadsegmente speichern. Wir unterscheiden dazu für einen Knoten  $v \in S_i$  drei Fälle:

*rot*: Das Pfadsegment von  $v$  enthält keinen 0-Knoten,

*gelb*: das Pfadsegment von  $v$  enthält einen 0-Knoten aber beim Übergang von  $S_{i-1}$  zu  $S_i$  ist kein neuer 0-Knoten hinzugekommen oder

*grün*: beim Übergang von  $S_{i-1}$  zu  $S_i$  ist zu dem Pfadsegment von  $v$  ein neuer 0-Knoten hinzugekommen.

Wir erkennen an der Folge  $S_0, S_1, \dots$  der Kodierungsbäume, daß der untersuchte Baum  $B$  einen Pfad besitzt, der unendlich oft nach links verzweigt, wenn es einen Knotennamen  $v$  gibt, der ab einem Index  $i$  zu allen Kodierungsbäumen mit größerem Index gehört und für den unendlich oft Fall *grün* eintritt.

Wir gehen nun zu einer formalen Beschreibung der Kodierungsbäume und des zuvor beschriebenen Verfahrens zur Berechnung der Folge  $S_0, S_1, \dots$  der Pfadsegmentierungsbäume zu einem gegebenen unendlichen  $\{0, 1\}$ -Baum über.

Sei  $B$  ein endlicher  $\{0, 1\}$ -Baum der Tiefe  $t$ .

Ein *Pfadsegmentierungsbaum*  $S$  (kurz S-Baum) zu  $B$  ist ein beschrifteter Baum über einer endlichen Knotennamenmenge  $\mathcal{V}$ , bei dem jeder Knoten genau zwei Nachfolger hat oder ein Blatt ist.

Die Beschriftung von  $S$  hat die Komponenten: anfang, ende, farbe und knoten.

Die Funktionen anfang und ende ordnen jedem Knoten  $v \in S$  einen Knoten aus  $B$  zu, wobei gilt:

$$\text{anfang}(v) = \text{ende}(v) \text{ oder } \text{ende}(v) \text{ ist ein Nachfahre von } \text{anfang}(v).$$

Für einen Nachfolger  $w$  von  $v$  gilt:  $\text{anfang}(w)$  ist ein Nachfolger von  $\text{ende}(v)$  in  $B$ . Für einen Knoten  $v \in S$  setzen wir  $\text{pfad}(v) := [\text{anfang}(v), \text{ende}(v)]$ ,  $\text{pfad}(v)$  ist das von  $v$  repräsentierte Pfadsegment des bereinigten Baums  $B'$  zu  $B$ . Für einen Knotennamen  $v \notin S$  bezeichne  $\text{pfad}(v)$  das leere Pfadsegment.

Die Funktion farbe ordnet jedem  $v \in \mathcal{V}$  ein Element aus  $\{\text{rot}, \text{gelb}, \text{grün}\}$  zu. Dabei gilt:

$$\text{farbe}(v) \in \{\text{gelb}, \text{grün}\} \Leftrightarrow \text{pfad}(v) \text{ enthält einen 0-Knoten.}$$

Die Funktion knoten ordnet jedem  $v \in S$  eine Teilmenge der Knoten der Tiefe  $t$  von  $B$  zu, wobei gilt:

- $\text{knoten}(v) \neq \emptyset$ ,
- ist  $v$  die Wurzel von  $S$ , so gilt:  $\text{knoten}(v)$  ist die Menge der Knoten der Tiefe  $t$  von  $B$ ,
- $|\text{knoten}(v)| = 1 \Leftrightarrow v$  ist ein Blatt von  $S$ ,
- ist  $v$  ein Blatt, so gilt:  $\text{knoten}(v) = \{\text{ende}(v)\}$  und
- sind  $v_1, v_2$  Nachfolger von  $v$ , so gilt:  $\text{knoten}(v) = \text{knoten}(v_1) \dot{\cup} \text{knoten}(v_2)$ .

Wir geben nun eine Pseudocodebeschreibung des oben angedeuteten Verfahrens an, das zu einem unendlichen  $\{0, 1\}$ -Baum  $B$  mit beschränkter Breite  $b$  eine Folge  $S_0, S_1, \dots$  von S-Bäumen zu  $B_0, B_1, \dots$  über  $\mathcal{V}_b$  bestimmt, wobei  $\mathcal{V}_b = \{1, \dots, 3b - 1\}$ . Für die Beschriftungskomponente Knoten von  $S_i$  schreiben wir  $\text{knoten}_i$ , die übrigen Komponenten der Beschriftung bezeichnen wir entsprechend.

Der Baum  $S_0$  bestehe nur aus einem Knoten, wir wählen als Knotennamen  $1 \in \mathcal{V}_b$  und wir setzen:  $\text{knoten}_0(1) := \{r\}$ ,  $\text{anfang}_0(1) := \text{ende}_0(1) := r$ , wobei  $r$  die Wurzel von  $B$  sei. Falls  $r = 1$  gilt, setzen wir  $\text{farbe}_0(1) := \text{rot}$ , ansonsten setzen wir  $\text{farbe}_0(1) := \text{grün}$ . Der Baum  $S_0$  ist offensichtlich ein S-Baum zu  $B_0$  über  $V_b$ .

Für  $i \geq 0$  bestimmen wir  $S_{i+1}$  aus  $S_i$  nach dem folgenden Verfahren:

### {Kopieren}

- 1:  $S'_i := S_i$
- 2: für jedes  $v \in S_i$
- 3:     wenn  $\text{farbe}_i(v) = \text{grün}$
- 4:          $\text{farbe}'_i(v) := \text{gelb}$

### {Erweitern}

- 5: für jedes  $v \in S_i$
- 6:      $\text{knoten}'_i(v) := \{y \mid y \text{ ist Nachfolger von } x \text{ für ein } x \in \text{knoten}_i(v)\}$
- 7: für jedes Blatt  $v \in S_i$
- 8:     wenn  $\text{knoten}'_i(v)$  einen 0-Knoten  $y$  enthält
- 9:         füge neuen Nachfolger  $w$  von  $v$  zu  $S'_i$  hinzu
- 10:          $\text{knoten}'_i(w) := \{y\}$ ;  $\text{farbe}'_i(w) := \text{grün}$
- 11:          $\text{anfang}'_i(w) := \text{ende}'_i(w) := y$
- 12:     wenn  $\text{knoten}'_i(v)$  einen 1-Knoten  $y$  enthält
- 13:         füge neuen Nachfolger  $w$  von  $v$  zu  $S_i$  hinzu
- 14:          $\text{knoten}'_i(w) := \{y\}$ ;  $\text{farbe}'_i(w) := \text{rot}$
- 15:          $\text{anfang}'_i(w) := \text{ende}'_i(w) := y$

### {Ausdünnen}

- 16: entferne alle Knoten  $v$  aus  $S'_i$  mit  $\text{knoten}'_i(v) = \emptyset$

### {Verdichten}

- 17: solange es noch Knoten in  $S'_i$  gibt, die genau einen Nachfolger haben
- 18:     wähle einen Knoten  $v \in S'_i$  aus, der genau einen Nachfolger  $w$  hat
- 19:     wenn  $\text{farbe}'_i(w) \in \{\text{grün}, \text{gelb}\}$
- 20:          $\text{farbe}'_i(v) := \text{grün}$
- 21:          $\text{ende}'_i(v) := \text{ende}'_i(w)$
- 22:     mache Nachfolger von  $w$  zu Nachfolgern von  $v$
- 23:     entferne  $w$  aus  $S'_i$
- 24:  $S_{i+1} := S'_i$

**Erläuterungen** Wird ein Knoten  $w$  aus  $S'_i$  entfernt, so werde  $\text{farbe}'_i(w) := \text{rot}$  gesetzt. Im Erweiterungsschritt werden neue Knoten zu  $S'_i$  hinzugefügt. Da der untersuchte

Baum die Breite  $b$  hat, werden höchstens  $b$  neue Knoten hinzugefügt. Aufgrund der an Segmentierungsbäume gestellte Bedingung hat  $S_i$  höchstens  $b$  Blätter. Nach Bemerkung (4.2) hat  $S_i$  somit höchstens  $2b - 1$  Knoten. Es gibt also für den Erweiterungsschritt genügend freie Knotennamen in  $\mathcal{V}_b$ . Um eine eindeutige Folge von Segmentierungsbäumen zu erhalten, werden bei den Schritten 7 und 18 die Knoten in der Reihenfolge ihrer Knotennamen behandelt. Bei den Schritten 9 und 13 werde der jeweils kleinste freie Knotenname verwendet.

Mit Hilfe des folgenden Lemmas ergibt sich, daß die nach dem obigen Verfahren bestimmte Folge  $S_0, S_1, \dots$  eine Folge von Pfadsegmentierungsbäumen zu  $B_0, B_1, \dots$  ist.

**Lemma 4.1** Sei  $B$  ein unendlicher  $\{0, 1\}$ -Baum. Sei  $V_i$  die Knotenmenge der  $i$ -ten Ebene von  $B$ . Sei  $S_0, S_1, \dots$  die Folge von Bäumen gemäß obigem Verfahren zu  $B$ . Sei  $i \in \mathbb{N}$ . Sei  $v \in S_i$ . Dann gelten die folgenden Aussagen:

- i)  $1 \in S_i$  und  $\text{knoten}_i(1) = V_i$
- ii) Sind  $v_1$  und  $v_2$  die Nachfolger von  $v$  in  $S_i$ , so gilt:  
 $\text{knoten}_i(v) = \text{knoten}_i(v_1) \dot{\cup} \text{knoten}_i(v_2)$ .
- iii) Ist  $x \in \text{knoten}_i(v)$ , so gilt:  $\text{ende}_i(v) = x$  oder  $x$  ist Nachfahre von  $\text{ende}_i(v)$ .
- iv) Sei  $y \in B$  von der Form  $y = x0$ . Ist  $y \in \text{pfad}_i(v)$  und  $y \notin \text{pfad}_{i-1}(v)$ , so gilt:  
 $\text{farbe}_i(v) = \text{grün}$ .
- v) Sei  $\text{farbe}_i(v) \in \{\text{gelb}, \text{grün}\}$ . Ist  $v \in S_{i+1}$ , so gilt:  $\text{farbe}_{i+1}(v) \in \{\text{gelb}, \text{grün}\}$ .
- vi) Sei  $x \in \text{knoten}_i(v)$ . Sei  $N$  die Menge der Nachfolger von  $x$  in  $B$ . Ist  $v \in S_{i+1}$ , so gilt:  $N \subset \text{knoten}_{i+1}(v)$ .
- vii) Ist  $\text{farbe}_i(v) = \text{grün}$ , so gilt:  $|\text{pfad}_i(v)|_0 > |\text{pfad}_{i-1}(v)|_0$ .

**Beweis** Für  $i = 0$  gelten die Aussagen i)-vii) offensichtlich. Seien die Aussagen gültig bis zu einem  $i \in \mathbb{N}$ . Dann gelten die Punkte i), ii), v) und vi) offensichtlich auch für  $i + 1$ .

Zu iii): Nach dem Erweiterungsschritt gilt für die neu hinzugekommenen Blätter  $v \in S'_i$ :  $\text{ende}'_i(v) = x$  und  $\text{knoten}'_i(v) = \{x\}$ . Für einen inneren Knoten  $v \in S'_i$  gilt offenbar für ein  $x \in \text{knoten}'_i(v)$ :  $x$  ist Nachfahre von  $\text{ende}_i(v) = \text{ende}'_i(v)$ . Übernimmt ein Knoten  $v$  beim Verdichten von  $S'_i$  einen Nachfolger  $w$ , so galt vorher  $\text{knoten}'_i(v) = \text{knoten}'_i(w)$ . Da zuvor für alle  $x \in \text{knoten}'_i(w)$  galt:  $\text{ende}'_i(w) = x$  oder  $x$  ist Nachfahre von  $\text{ende}'_i(w)$  und in Schritt 21  $\text{ende}'_i(v) := \text{ende}'_i(w)$  gesetzt wird, bleibt die Bedingung durch das Verdichten erhalten.

Zu iv): Wenn  $v \notin S_i$  gilt, dann muß  $v$  ein neues Blatt von  $S_{i+1}$  sein und es muß gelten  $\text{knoten}_{i+1} = \{y\}$  und daher wegen der Schritte 8-10 ist  $\text{farbe}_{i+1}(v) = \text{grün}$ . Wenn

$v \in S_i$  gilt, dann muß  $v$  im Verlauf des Übergangs von  $S_i$  zu  $S_{i+1}$  einen Knoten  $w$  übernommen haben mit:  $y \in \text{pfad}'_i(w)$ . Der Knoten  $y$  wird beim Übergang zu  $S_{|y|-1}$  in einem grünen Blatt in den Schritten 8-10 in  $S_{|y|-2}$  aufgenommen. Der Knoten, zu dessen Pfad  $y$  gehört ist in jedem der folgenden Bäume offensichtlich gelb oder grün. Daher gilt bei der Übernahme von  $w$  durch  $v$ :  $\text{farbe}'_i(v) \in \{\text{gelb}, \text{grün}\}$  und somit gilt  $\text{farbe}_{i+1}(v) = \text{grün}$ .

Zu vii): Ist  $v$  ein neuer Knoten von  $S_{i+1}$ , dann ist  $v$  ein Blatt. Damit besteht  $\text{pfad}_{i+1}(v)$  aus genau einem 0-Knoten und die Aussage gilt offensichtlich, da für Knotennamen  $w$ , die nicht in  $S_i$  liegen, per Definition  $\text{pfad}_i(w)$  der leere Pfad ist. Ist  $v \in S_i$ , so muß  $v$  beim Verdichten einen Nachfolger  $w$  übernommen haben mit:  $\text{farbe}'_i(w) \in \{\text{grün}, \text{gelb}\}$ . Es muß also für ein  $j \leq i + 1$  beim Übergang zu  $S_j$  ein neues Blatt  $u$  zu  $S_j$  hinzugekommen sein, daß ein Nachfolger von  $v$  war, mit  $\text{farbe}'_{j-1} = \text{grün}$  nach den Schritten 8-10 und dessen 0-Knoten in einer Folge von Übernahmen zu  $\text{pfad}'_i(w)$  hinzugekommen ist.  $\square$

Mit Hilfe von Lemma (4.1) beweisen wir den folgenden Satz. Der Satz bietet uns die Möglichkeit dieses Baumuntersuchungsverfahrens in Form eines deterministischen Rabin-Automaten zu simulieren.

**Satz 4.1** Sei  $B$  ein unendlicher  $\{0, 1\}$ -Baum mit beschränkter Breite  $b$ . Sei  $S_0, S_1, S_2, \dots$  die Folge der Pfadsegmentierungsbäume zu  $B$  über  $\mathcal{V}_b = \{1, \dots, 3b - 1\}$ . Dann sind die folgenden Aussagen äquivalent:

- i) Es gibt einen Pfad  $\pi$  in  $B$ , der unendlich oft nach links verzweigt.
- ii) Es gibt einen Knotennamen  $v \in \mathcal{V}_b$  mit  $\text{farbe}_i(v) = \text{grün}$  für unendlich viele  $i \in \mathbb{N}$  und  $\text{farbe}_i(v) = \text{rot}$  für nur endlich viele  $i \in \mathbb{N}$ .

**Beweis** i) $\Rightarrow$ ii): Sei  $\pi$  ein Pfad in  $B$ , der unendlich oft nach links verzweigt. Wir suchen einen Knotennamen  $v \in \mathcal{V}_b$ , der die folgenden Bedingungen erfüllt:

- (1)  $\exists i \in \mathbb{N} \forall j \geq i (v \in S_j \text{ und } \pi(j) \in \text{knoten}_j(v))$  und
- (2)  $v$  übernimmt unendlich oft einen Nachfolger.

Erfüllt ein Knotenname  $v \in \mathcal{V}_b$  diese Bedingungen, so gibt es ein  $i_v \in \mathbb{N}$  derart, daß für alle  $j \geq i_v$  gilt:  $v \in S_j$  und  $\pi(j) \in \text{knoten}_j(v)$ , d.h. nach Lemma (4.1) iii)  $\pi(j) = \text{ende}_j(v)$  oder  $\pi(j)$  ist Nachfahre von  $\text{ende}_j(v)$ , d.h.  $\text{ende}_j(v)$  liegt auf  $\pi$ .

Nach Bedingung (2) übernimmt  $v$  unendlich oft einen Nachfolger, d.h. es gilt

$$\lim_{j \rightarrow \infty} \text{tiefe}(\text{ende}_j(v)) = \infty.$$

Es gilt also:

- (\*) Für jedes  $i \geq i_v$  gibt es ein  $j \geq i$  so, daß  $\pi(i)$  auf  $\text{pfad}_j(v)$  liegt.

Wir zeigen nun, daß  $\text{farbe}_i(v) = \text{grün}$  für unendlich viele  $i \in \mathbb{N}$  und  $\text{farbe}_i(v) = \text{rot}$  nur für endlich viele  $i \in \mathbb{N}$  gilt. Sei dazu zunächst  $j \geq i_v$ . Nach Voraussetzung gibt es ein  $j' > j$  mit  $\pi(j')$  ist ein 0-Knoten. Nach (\*) gibt es ein minimales  $j'' \geq j'$  mit  $\pi(j'')$  liegt auf  $\text{pfad}_{j''}(v)$ , nach Lemma (4.1) iv) gilt  $\text{farbe}_{j''}(v) = \text{grün}$ . Es gilt also für unendlich viele  $i \in \mathbb{N}$ :  $\text{farbe}_i(v) = \text{grün}$ . Ist  $\text{farbe}_j(v) = \text{grün}$  für ein  $j \geq i_v$ , so folgt mit Lemma (4.1) v) für alle  $j' \geq j$ :  $\text{farbe}_{j'} \in \{\text{grün}, \text{gelb}\}$ . Es gilt also nur für endlich viele  $i \in \mathbb{N}$ :  $\text{farbe}_i(v) = \text{rot}$ .

Es bleibt zu zeigen, daß ein Knotenname  $v \in \mathcal{V}_b$  existiert, der die Bedingungen (1) und (2) erfüllt. Der Knotenname 1 erfüllt die Bedingung (1), siehe Lemma (4.1) i). Übernimmt 1 unendlich oft einen Nachfolger, so ist 1 der gesuchte Knotenname. Übernimmt 1 nur endlich oft einen Nachfolger, so gibt es ein  $i \in \mathbb{N}$  derart, daß 1 in  $S_i$  zwei Nachfolger  $v_1$  und  $v_2$  hat, für die für alle  $j \geq i + 1$  gilt  $v_1, v_2 \in S_j$ . Nach Lemma (4.1) ii) gilt entweder  $\pi(i) \in \text{knoten}_i(v_1)$  oder  $\pi(i) \in \text{knoten}_i(v_2)$ . Es gelte o.B.d.A.  $\pi(i) \in \text{knoten}_i(v_1)$ . Mit Lemma (4.1) vi) folgt  $\pi(j) \in \text{knoten}_j(v_1)$  für alle  $j \geq i$ . Übernimmt Knoten  $v_1$  unendlich oft einen Nachfolger, so ist  $v_1$  der gesuchte Knotenname. Übernimmt  $v_1$  nur endlich oft einen Nachfolger, so fahren wir in derselben Weise fort. Da die Tiefe der Kodierungsbäume beschränkt ist, erhalten wir so einen Knoten  $v$ , der die Bedingungen (1) und (2) erfüllt.

ii) $\Rightarrow$ i): Sei  $v \in \mathcal{V}_b$  ein Knotenname für den für unendlich viele  $i \in \mathbb{N}$  gilt  $\text{farbe}_i(v) = \text{grün}$  und für den nur für endlich viele  $i \in \mathbb{N}$  gilt  $\text{farbe}_i(v) = \text{rot}$ . Es gibt also ein  $i_v \in \mathbb{N}$  so, daß für alle  $j \geq i_v$  gilt  $v \in S_j$ .

Seien  $j_0 < j_1 < j_2 \dots$  Indizes mit  $j_0 = i_v$  und  $\text{farbe}_{j_k}(v) = \text{grün}$  für  $k \geq 1$ . Für  $k \geq 0$  gilt:  $\text{ende}_{j_{k+1}}(v)$  ist ein Nachfahre von  $\text{ende}_{j_k}$  in  $B$ . Es gibt also einen unendlichen Pfad  $\pi$  in  $B$ , der die Knoten  $\text{ende}_{j_k}(v)$  für  $k \geq 0$  besucht.

Nach Lemma (4.1) vii) ist die Anzahl der Linksabzweigungen auf dem Pfadsegment  $\text{pfad}_{j_{k+1}}(v)$  mindestens um eins größer als die Anzahl der Linksabzweigungen auf dem Pfadsegment  $\text{pfad}_{j_k}(v)$ . Also gibt es auf dem Pfadsegment von der Wurzel von  $B$  nach  $\text{ende}_{j_k}(v)$  mindestens  $k$  Linksabzweigungen. Insgesamt zweigt  $\pi$  also unendlich oft nach links ab.  $\square$

Betrachten wir den Algorithmus zur Bestimmung des Kodierungsbaums  $S_{i+1}$  aus  $S_i$  genauer, so erkennen wir, daß die Komponenten anfang und ende nicht wesentlich sind, sie dienen nur der Vereinfachung der Argumentation im Beweis zu Satz (4.1). Wir können diese Komponenten weglassen.

Kennen wir die Beschriftung der Blätter eines S-Baums  $S$ , so ergeben sich die Werte  $\text{knoten}(v)$  für innere Knoten  $v$  durch sukzessive Vereinigung. Wir können uns bei der Komponente  $\text{knoten}$  der Beschriftung somit auf die Blätter beschränken. Dies werden wir bei der späteren Anwendung des Verfahrens berücksichtigen.

Wir haben hier das Verfahren der Pfadsegmentierung speziell für die Suche nach einem unendlich oft nach links verzweigenden Pfad in einem  $\{0, 1\}$ -Baum beschrieben. Durch entsprechende Abänderung der Definition von Segmentierungsbäumen und des Verfahrens zur Bestimmung des Folge der Segmentierungsbäume können wir dieses Ver-

fahren leicht in einem allgemeineren Rahmen nutzen: Dabei betrachten wir unendliche  $K$ -Bäume  $B$ , deren Knotenmenge in zwei Klassen  $K_0$  und  $K_1$  partitioniert ist. In diesem allgemeineren Fall suchen wir einen Pfad von  $B$ , der unendlich viele Knoten von  $K_0$  enthält. Das im folgenden Abschnitt beschriebene Verfahren geht von der allgemeineren Problemstellung aus.

### 4.3 Pfadverfolgung

Das Verfahren der Pfadverfolgung nach Emerson und Sistla untersucht unendliche  $K$ -Bäume  $B$  mit beschränkter Breite  $b$ . Dabei ist die Knotenmenge von  $B$  in zwei Klassen  $K_0$  und  $K_1$  unterteilt, d.h.  $B = K_0 \dot{\cup} K_1$ . Ziel ist es zu entscheiden, ob der Baum  $B$  einen Pfad enthält, auf dem unendlich viele Knoten aus  $K_0$  liegen.

Durch die beschränkte Breite  $b$  von  $B$  kann es in  $B$  höchstens  $b$  verschiedene unendliche Pfade geben. Jedoch kann es Pfadsegmente geben, die in Sackgassen führen, d.h. die nicht auf einem unendlichen Pfad liegen, diese Möglichkeit stellt sich als Hauptschwierigkeit heraus.

Es müssen sehr viele Informationen zu Pfadsegmenten gespeichert werden und die Verwaltung dieser Informationen läßt das Verfahren kompliziert werden. Eine Abwandlung dieses Verfahrens für  $\{0, 1\}$ -Bäume werden wir im nächsten Abschnitt darstellen. Es verfolgt dieselbe Idee aber nutzt die spezielle Struktur von  $\{0, 1\}$ -Bäumen aus, um die zu speichernden Informationen und den nötigen Verwaltungsaufwand zu verringern. Wir stellen das Verfahren hier nur auf einer informellen Ebene dar, um die Idee zu verdeutlichen.

Wie der Name des Verfahrens andeutet, wird versucht Pfade des gegebenen Baums  $B$  zu verfolgen. Für die Verfolgung der Pfade werden *Markierungen* (kurz Marker) verwendet. Jedem verfolgten Pfadsegment von  $B$  wird ein Marker zugeordnet. Der Baum  $B$  wird ebenenweise untersucht, d.h. im  $i$ -ten Schritt des Verfahrens werden diejenigen Pfadsegmente betrachtet, die von der Wurzel von  $B$  bis zur Ebene  $i$  von  $B$  führen, diese Pfadsegmente heißen *aktive Pfadsegmente*. Da  $B$  eine Breite von  $b$  hat, sind nur  $b$  verschiedene Marker notwendig, um alle aktiven Pfadsegmente zu repräsentieren.

Für einen Marker  $i$  gibt es  $b$  *Signalpaare*  $\langle i, 0 \rangle, \dots, \langle i, b-1 \rangle$ . Insgesamt gibt es also  $b^2$  solche Signalpaare. Ein Signalpaar  $\langle i, j \rangle$  besteht aus einem grünen und einem roten Signal. Die Bedeutung dieser Signalpaare wird später erläutert.

Beim Übergang von einer Ebene von  $B$  zur nächsten, gibt es für das durch einen Marker  $i$  repräsentierte Pfadsegment  $\mu_i$  drei Möglichkeiten:

1.  $\mu_i$  kann nicht in die nächste Ebene fortgesetzt werden,
2. es gibt genau einen Knoten der nächsten Ebene, durch den  $\mu_i$  fortgesetzt werden kann oder
3. es gibt mehrere Knoten durch die  $\mu_i$  erweitert werden kann.

Im Fall 3. müssen wir eine der möglichen Alternativen für die Erweiterung von  $\mu_i$  auswählen. Dabei können wir falsche Entscheidungen treffen, d.h. wir erweitern  $\mu_i$  durch einen Knoten derart, daß  $\mu_i$  nicht mehr zu einem unendlichen Pfad erweitert werden kann. Wüßten wir, welche der möglichen Folgeknoten auf einem unendlichen Pfad liegen, so kämen wir für jeden Marker  $i$  mit nur einem Signal  $\langle i, 0 \rangle$  aus. Wir müßten einen der Folgeknoten  $x_1, \dots, x_l$  von  $\mu_i$ , die auf einem unendlichen Pfad liegen, auswählen und  $\mu_i$  um diesen erweitern, z.B. den Knoten  $x_1$ . Für die anderen Folgeknoten, die auf einem unendlichen Pfad liegen, wählen wir noch freie Marker aus und diese beginnen mit der Verfolgung der Pfade, die von  $x_2, \dots, x_l$  ausgehen. Da es höchstens  $b$  verschiedene unendliche Pfade in  $B$  gibt, reichen dazu  $b$  Marker aus. Wir das Pfadsegment eines Markers  $i$  um einen Knoten aus  $K_0$  erweitert, so lösen wir das Signal  $\langle i, 0 \rangle$  aus. Dann gibt es genau dann einen Pfad von  $B$ , der unendlich viele Knoten aus  $K_0$  besucht, wenn ein Signal  $\langle i, 0 \rangle$  gibt, das unendlich oft ausgelöst wird.

Wir haben jedoch keine Möglichkeit, bei verschiedenen Erweiterungsmöglichkeiten eines Pfadsegments  $\mu_i$ , zu entscheiden, welche zu einem unendlichen Pfad gehört und welche nicht. Daher kann es sein, daß ein Pfadsegment  $\mu_i$  in eine Sackgasse gerät, da wir zuvor  $\mu_i$  falsch erweitert haben.

Beginnt der Marker  $i$  zu einem späteren Zeitpunkt mit der Verfolgung eines anderen Pfadsegments, so kann es geschehen, daß das Signal  $\langle i, 0 \rangle$  unendlich oft ausgelöst wird, obwohl die auslösenden  $K_0$ -Knoten nicht auf einen Pfad liegen.

Um diese Möglichkeit zu berücksichtigen, könnten wir aus den Signalen  $\langle i, 0 \rangle$  Signalkopiepaare machen, die aus einem grünen und einem roten Signal bestehen. Wann immer der Marker  $i$  in eine Sackgasse gerät, wird das rote Signal des Signalkopiepaares  $\langle i, 0 \rangle$  ausgelöst; besucht das zu  $i$  gehörende Pfadsegment einen Knoten aus  $K_0$ , so wird das grüne Signal ausgelöst. Wird zu einem Marker  $i$  unendlich oft das grüne Signal ausgelöst und nur endlich oft das rote Signal, so gibt es einen Zeitpunkt, von dem an  $i$  nie wieder in eine Sackgasse gerät und danach noch unendlich oft einen  $K_0$ -Knoten besucht, siehe Abbildung (8a). Wird andererseits das rote Signal zum Marker  $i$  unendlich oft ausgelöst, so wissen wir, daß der Marker  $i$  jeweils nur endliche Pfadsegmente verfolgt hat, siehe Abbildung (8b).

Gehen wir auf diese Art vor, so kann es sein, daß wir einen Pfad, der unendlich viele  $K_0$ -Knoten enthält übersehen. Siehe Abbildung (9a).

Wir dürfen die Informationen, die ein Marker während der Verfolgung eines Pfadsegments sammelt, nicht vergessen, wenn er in eine Sackgasse gerät, sondern wir müssen die verwertbaren Informationen weiter nutzen. Hierzu führen wir den Mechanismus des *Übernehmens* ein. Das bedeutet, wenn ein Marker  $i$  in eine Sackgasse gerät dann versucht er, so lange auf seinem Pfadsegment zurückzugehen, bis er ein Pfadsegment findet, welches von seinem abzweigt und das in die nächste Ebene weitergeführt werden kann.

Als Beispiel betrachten wir Abbildung (9b). Wenn  $\mu_1$  in eine Sackgasse gerät, übernimmt er das von Marker 2 verfolgte Pfadsegment, das von  $\mu_1$  zuvor abgezweigt ist. Der Marker 2 ist danach keinem Pfadsegment zugeordnet. Dies zeigen wir dadurch an, daß

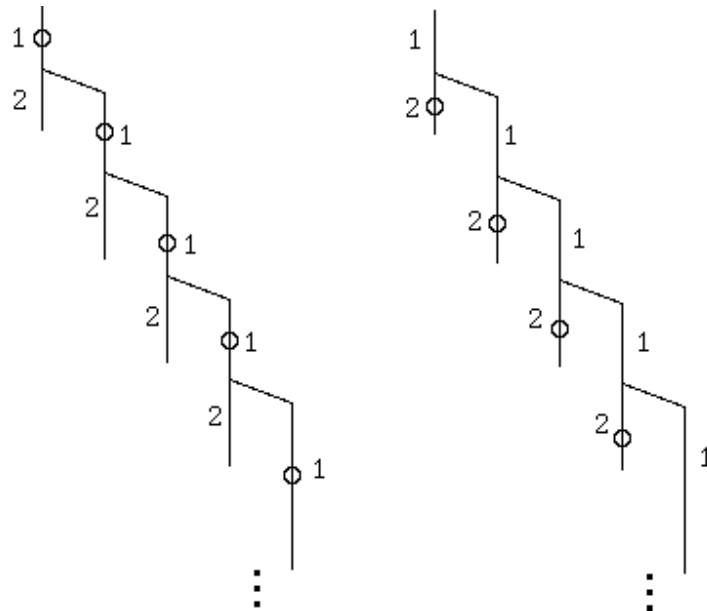


Abbildung 8: Erläuterung der Funktionsweise der Signalpaare. Knoten aus  $K_0$  sind durch Kreise gekennzeichnet. a) Das grüne Signal des Paares  $\langle 1, 0 \rangle$  wird unendlich oft ausgelöst. b) Das rote Signal des Paares  $\langle 2, 0 \rangle$  wird unendlich oft ausgelöst.

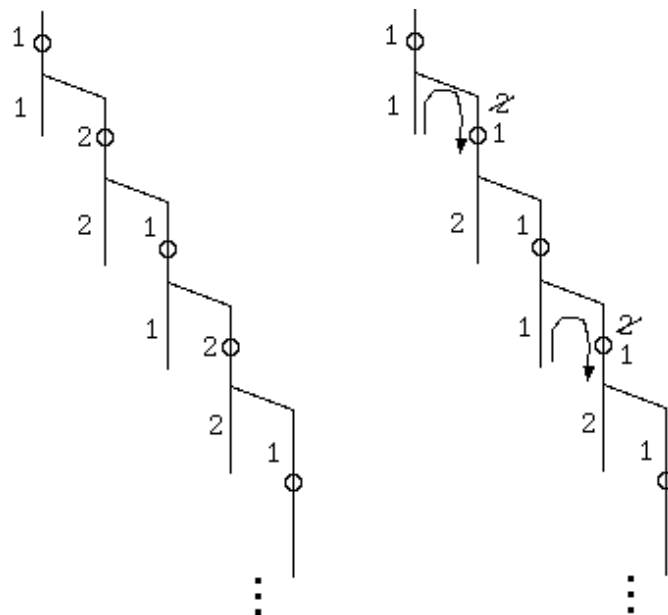


Abbildung 9: a) Für beide Paare  $\langle 1, 0 \rangle$  und  $\langle 2, 0 \rangle$  wird das rote Signal unendlich oft ausgelöst. b) Mechanismus des Übernehmens von Pfadsegmenten. Der Marker 1 übernimmt Pfadsegmente von Marker 2.

das rote Signal der beiden Signalpaar  $\langle 2, 0 \rangle$  und  $\langle 2, 1 \rangle$  ausgelöst wird. Da Marker 1 sein Pfadssegment  $\mu_1$  nicht auf direktem Weg fortsetzen konnte, lösen wir das rote Signal des Signalpaares  $\langle 1, 0 \rangle$  aus. Das rote Signal des Paares  $\langle 1, 0 \rangle$  dient dazu, ein unendlich oft ausgelöstes grünes Signal des Paares  $\langle 1, 0 \rangle$  nicht falsch interpretieren. Weil auf dem gekennzeichneten Pfadssegment, das nun zu  $\mu_1$  gehört, ein  $K_0$ -Knoten liegt, lösen wir das grüne Signal des Paares  $\langle 1, 1 \rangle$  aus.

Allgemein können wir für den Fall eines unendlichen  $K$ -Baums der Breite 2 die folgenden Fälle für den Marker 1 unterscheiden:

1. Ab einer bestimmten Tiefe setzt Marker 1 sein Pfadssegment  $\mu_1$  immer direkt fort. Dann wird nie wieder das rote Signal des Paares  $\langle 1, 0 \rangle$  ausgelöst. Dann enthält der von Marker 1 verfolgte unendliche Pfad genau dann unendlich viele  $K_0$ -Knoten, wenn das grüne Signal des Paares  $\langle 1, 0 \rangle$  unendlich oft ausgelöst wird.
2. Marker 1 gerät unendlich oft in eine Sackgasse, d.h. es wird unendlich oft das rote Signal des Paares  $\langle 1, 0 \rangle$  ausgelöst. Dann enthält der von Marker 1 verfolgte Pfad genau dann unendlich viele  $K_0$ -Knoten, wenn das grüne Signal des Paares  $\langle 1, 1 \rangle$  unendlich oft ausgelöst wird.

Um die Verallgemeinerung des eben beschriebenen Prinzips der Signalpaare zu verdeutlichen, stellen wir die Entwicklung eines Pfadssegments  $\mu_i$  auf besondere Weise dar. Zunächst gehen wir zu einer horizontalen Darstellung über, d.h. Bäume verlaufen von links nach rechts und nicht – wie üblich – von oben nach unten. Betrachte zum folgenden die Abbildung (10). In Abbildung (10a) gerät Marker 1 in eine Sackgasse und übernimmt das Pfadssegment von Marker 2. Wir interpretieren diese Übernahme als Überspringen des Pfadssegments, das in eine Sackgasse geführt hat und stellen dies wie in (10a) rechts angedeutet dar. Eine Übernahme dieser Art nennen wir eine Übernahme erster Stufe. Gerät das Pfadssegment  $\mu_1$  wiederum in eine Sackgasse und übernimmt Marker 1 ein Pfadssegment, so gibt es dafür zwei Möglichkeiten:

1. das übernommene Pfadssegment zweigt nach dem Knoten  $v_1$  von  $\mu_1$  ab oder
2. das übernommene Pfadssegment zweigt vor dem Knoten  $v_2$  von  $\mu_1$  ab.

Im ersten Fall ergibt sich wiederum eine Übernahme der Stufe 1, im zweiten Fall nennen wir die Übernahme eine Übernahme der Stufe 2, denn es mußte eine vorherige Übernahme der Stufe 1 zurückgenommen werden.

Im allgemeinen Fall heißt eine Übernahme Übernahme der Stufe  $n$ , falls die maximale Übernahmestufe, die für diese Übernahme zurückgenommen werden mußte,  $n - 1$  ist, siehe Abbildung (10). Aus der graphischen Darstellung erkennen wir, daß es im Baum eine Ebene der Breite mindestens  $n + 1$  geben muß, wenn ein Marker eine Übernahme der Stufe  $n$  vollzogen hat. Hat der untersuchte Baum  $B$  die Breite  $b$ , kann es also höchstens Übernahmen der Stufe  $b - 1$  geben.

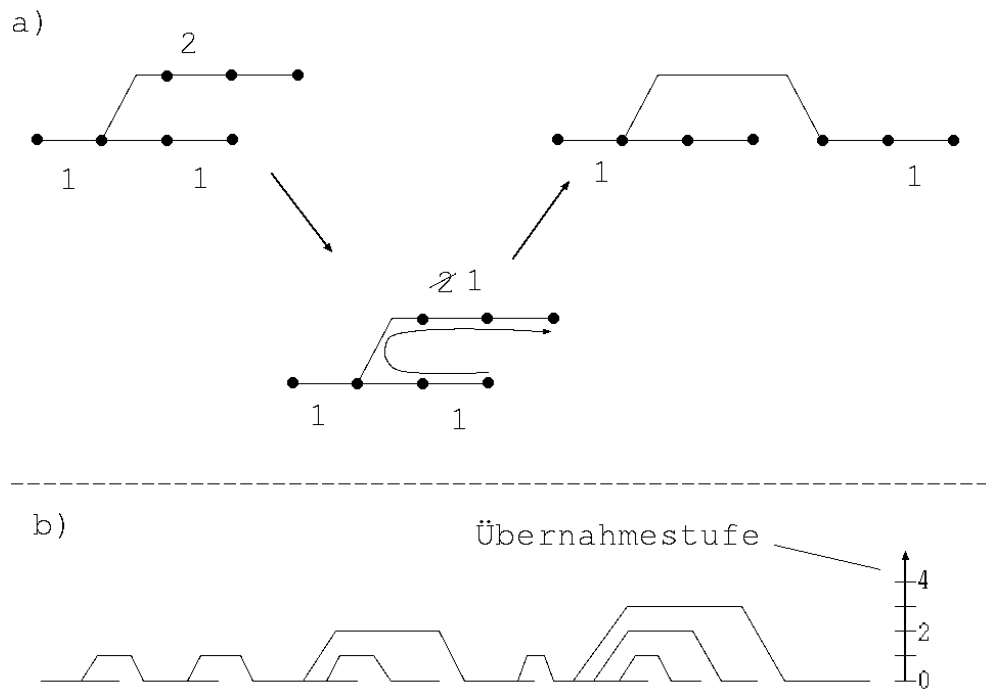


Abbildung 10: Alternative Darstellung der Übernahme eines Pfadsegments zur Verdeutlichung des Begriffs *Übernahmestufe*

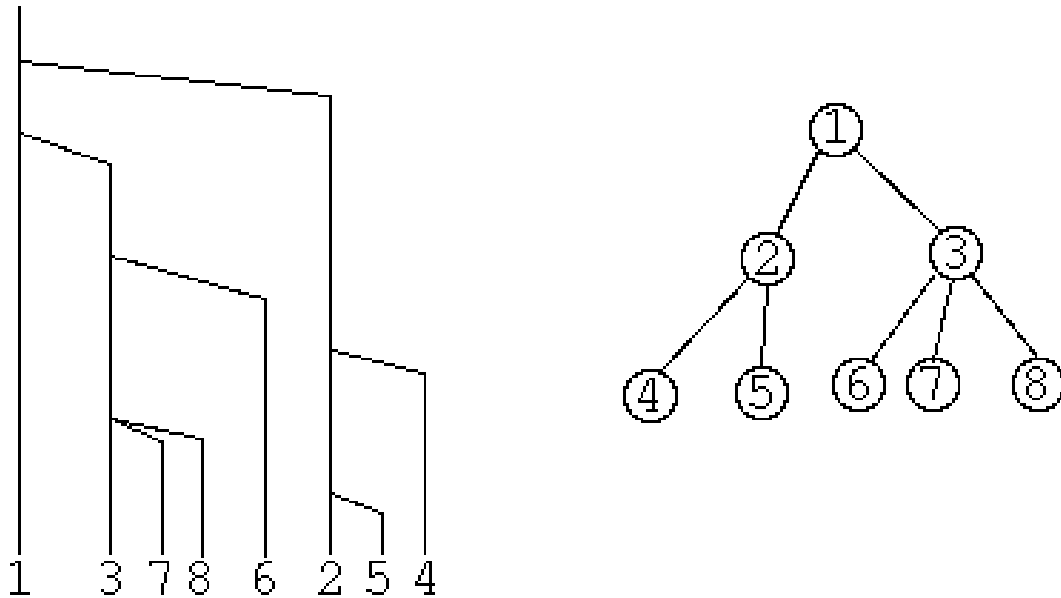


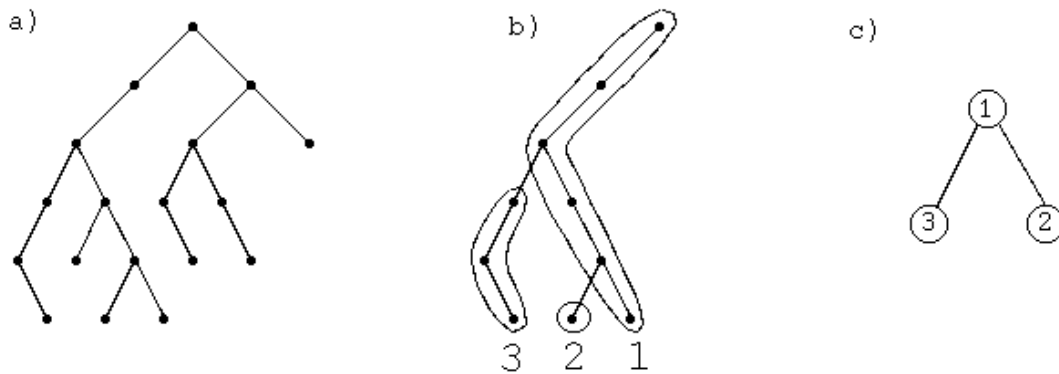
Abbildung 11: Beispielverzweigungsbaum

Wie es im einführenden Beispiel angedeutet wurde, wird einem Marker  $i$  zu jeder Übernahmestufe  $k$  ein Signalpaar  $\langle i, k \rangle$  zugeordnet. Verfolgt ein Marker  $i$  einen unendlichen Pfad, so gibt es eine maximale Übernahmestufe, die er dabei verwendet. Dabei gibt es eine maximale Übernahmestufe, die er unendlich oft verwendet, bezeichne diese mit  $k^*$ . Das Verfahren der Signalauslösung wird so von dem für zwei Signalpaare erweitert, daß gilt:

- Nach der letzten Übernahme von Marker  $i$  einer größeren Stufe als  $k^*$  wird das rote Signal des Paares  $\langle i, k^* \rangle$  nicht mehr ausgelöst und
- das grüne Signal des Paares wird genau dann unendlich oft ausgelöst, wenn auf dem von  $i$  verfolgten Pfad unendlich viele  $K_0$ -Knoten liegen.

Um den Mechanismus des Übernehmens von Markern zu realisieren führen wir den *Verzweigungsbaum* ein. Im Verzweigungsbaum gibt es zu jedem Marker  $i$  einen Knoten  $v_i$ , in dem Informationen zu diesem Marker gespeichert werden. Ist der Knoten  $v_i$  Nachfolger von  $v_j$ , so bedeutet dies, daß das Pfadsegment zu Marker  $i$  vom Pfadsegment zu Marker  $j$  abgezweigt ist. Die Ordnung der Nachfolger eines Knotens  $v_j$  gibt die Reihenfolge an, in der die Pfadsegmente der Nachfolger vom Pfadsegment zu  $j$  abgezweigt sind. Siehe Abbildung (11).

Für eine genauere und formale Beschreibung der Verwaltung der Informationen und des Mechanismus der Übernahme mit Hilfe des Verzweigungsbaums sei auf [ES84] verwiesen.

Abbildung 12: Kodierung eines endlichen  $\{0,1\}$ -Baums durch Pfadverfolgung

#### 4.4 Pfadverfolgung für $\{0,1\}$ -Bäume

Das Prinzip der Pfadverfolgung kann für unendliche  $\{0,1\}$ -Bäume mit beschränkter Breite, bei denen nach einem unendlich oft nach links verzweigenden Pfad gesucht wird, wesentlich vereinfacht werden. Ein Marker  $i$  verfolgt dabei solange wie möglich Rechtsabzweigungen, existiert keine Rechtsabzweigung, so übernimmt er einen seiner Nachfahren  $j$ . Da  $i$  zuvor wenn möglich Rechtsabzweigungen verfolgt hat, bedeutet das Übernehmen insbesondere, daß auf dem Pfad von  $i$  nun links abgezweigt wurde. Durch die beschränkte Breite des untersuchten Baums ergibt sich, daß ein Marker, der unendlich oft die Pfadsegmente anderer Marker übernimmt, schließlich einen Pfad repräsentiert, der unendlich oft nach links verzweigt. Die Anzahl der Informationen, die wir zu einem Marker speichern müssen, ist Vergleich zum Verfahren des vorigen Abschnitts geringer.

In Abbildung (12a) sei der Baum  $B_5$  eines zu untersuchenden unendlichen  $\{0,1\}$ -Baums dargestellt. Der bereinigte Baum  $B'_5$ , der nur aus den Knoten von  $B_5$  besteht, die auf einem unendlichen Baum liegen können, ist in Abbildung (12b) wiedergegeben.

Um den Baum  $B'_5$  zu kodieren, ordnen wir zunächst die Pfade von  $B'_5$ , die von der Wurzel zu einem Blatt führen, von rechts nach links. Die Ordnung dieser Pfade stimmt mit der Ordnung der zu den Pfaden gehörenden Blättern überein. In Abbildung (12b) sind die Blätter gemäß dieser Ordnung numeriert. Die zu diesen Blättern gehörenden Pfade seien dementsprechend mit  $\pi_1, \pi_2$  und  $\pi_3$  bezeichnet. Wir bilden nun Pfadsegmente  $\mu_1, \mu_2$  und  $\mu_3$  wie folgt:

- $\mu_1 := \pi_1$ ,
- für  $i = 1, 2$  sei  $\mu_i$  das Pfadsegment von  $\pi_i$ , das mit keinem  $\mu_k$  für  $k < i$  einen gemeinsamen Knoten hat.

In Abbildung (12b) sind die sich so ergebenden Pfadsegmente  $\mu_1, \mu_2$  und  $\mu_3$  umrandet. Die Abhängigkeiten unter den  $\mu_i$  stellen wir in einem geordneten Baum  $F_5$  dar, siehe Abbildung (12c). Die Ordnung des Baums  $F_5$  gibt die Ordnung der Pfade von  $B'_5$  wieder.

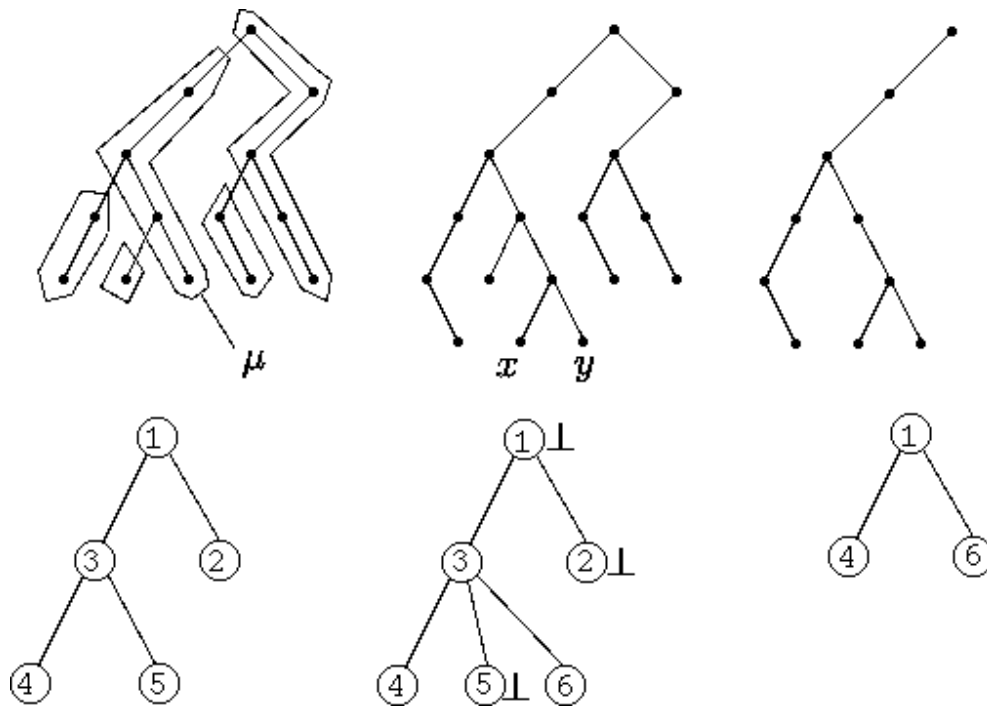


Abbildung 13: Übergang vom Pfadverfolgungsbaum  $F_4$  zum Baum  $F_5$

Zu der Folge  $B'_0, B'_1, \dots$  der bereinigten Teilbäume von  $B$  bilden wir die Folge  $F_0, F_1, \dots$  der dazugehörigen Kodierungsbäume. Dabei bestimmen wir  $F_{i+1}$  aus  $F_i$ . Um das allgemeine Vorgehen beim Übergang von  $F_i$  zu  $F_{i+1}$  zu erläutern, betrachten wir den Übergang von  $F_4$  zu  $F_5$  zum obigen Beispielbaum, siehe Abbildung (13). Das zum Knoten 3 in  $F_4$  gehörende Pfadsegment  $\mu$  kann durch den Knoten  $x$  oder den Knoten  $y$  erweitert werden. Gemäß der Strategie wenn möglich Rechtsabzweigungen zu verfolgen, repräsentiert der Knoten 3 nun zunächst das um  $y$  erweiterte Pfadsegment  $\mu$ . Für das beim Knoten  $x$  beginnende Pfadsegment führen wir einen neuen Knoten 6 ein. Wir machen 6 zum rechtesten Nachfolger von 3, da  $x$  der Anfangsknoten des bis hierhin rechtesten von  $\mu$  abzweigenden Pfadsegments ist.

Die mit  $\perp$  markierten Knoten stehen für Pfadsegmente von  $B'_4$ , die nicht durch einen Knoten der folgenden Ebene erweitert werden können. Betrachten wir als Beispiel den Wurzelknoten. Er soll den rechtesten Pfad von  $B'_5$  repräsentieren. Der rechteste Pfad von  $B'_5$  endet im Knoten  $y$ , da die Blätter von  $B'_4$ , die weiter rechts liegen als 3 keine Nachfolger besitzen. Durchlaufen wir  $F'_4$  gemäß preorder Durchlaufstrategie, wobei wir die Nachfolger eines Knotens von rechts nach links durchlaufen, ist 3 der erste Knoten, der nicht mit  $\perp$  markiert ist. Der Wurzelknoten übernimmt den Knoten 3 und das zu ihm gehörende Pfadsegment. Die mit  $\perp$  markierten Knoten, bei denen alle Nachfahren auch mit  $\perp$  markiert sind, können kein Pfadsegment übernehmen und fallen aus  $F'_4$

heraus, siehe Abbildung (13).

Wir gehen nun zu einer formalen Beschreibung der beschriebenen Kodierungsbäume und des Verfahrens zur Bestimmung der Folge der Kodierungsbäume zu einem gegebenen unendlichen  $\{0, 1\}$ -Baum über.

Sei  $B$  ein endlicher  $\{0, 1\}$ -Baum der Tiefe  $t$ . Ein Pfadverfolgungsbaum  $F$  (kurz F-Baum) zu  $B$  ist ein geordneter beschrifteter Baum über einer endlichen Knotennamensmenge  $\mathcal{V}$ .

Die Beschriftung von  $C$  hat die Komponenten *anfang*, *ende* und *farbe*. Die Komponenten *anfang* und *ende* ordnen jedem Knoten  $v \in F$  einen Knoten von  $B$  zu, wobei gilt:

- *ende* ist eine bijektive Abbildung zwischen den Knoten von  $F$  und den Blättern der Tiefe  $t$  von  $B$  und
- $\text{anfang}(v) = \text{ende}(v)$  oder  $\text{ende}(v)$  ist ein Nachfahre von  $\text{anfang}(v)$ .

Für einen Nachfolger  $w$  von  $v$  gilt:  $\text{anfang}(w)$  ist ein Nachfahre von  $\text{anfang}(v)$ . Für einen Knoten  $v \in C$  setzen wir  $\text{pfad}(v) := [\text{anfang}(v), \text{ende}(v)]$ ,  $\text{pfad}(v)$  ist das von  $v$  repräsentierte Pfadsegment des bereinigten Baums  $B'$  zu  $B$  und entspricht den oben beschriebenen  $\mu$ .

Die Ordnung der Knoten eines F-Baums soll der Ordnung der repräsentierten Pfadsegmente entsprechen. Wir fordern deshalb für zwei verschiedene Knoten  $v, w$  von  $F$ :

$$v <_{\text{post}} w \Leftrightarrow \text{ende}(v) <^{01} \text{ende}(w).$$

Wir geben nun eine Pseudocodebeschreibung des oben angedeuteten Verfahrens an, das zu einem unendlichen  $\{0, 1\}$ -Baum  $B$  mit beschränkter Breite  $b$  eine Folge  $F_0, F_1, \dots$  von F-Bäumen zu  $B_0, B_1, \dots$  über  $\mathcal{V}_b$  bestimmt, wobei  $\mathcal{V}_b = \{1, \dots, 2b\}$ . Für die Beschriftungskomponente *ende* von  $F_i$  schreiben wir  $\text{ende}_i$ , die übrigen Komponenten der Beschriftung bezeichnen wir entsprechend.

Der Baum  $F_0$  bestehe nur aus einem Knoten, wir wählen als Knotennamen  $1 \in \mathcal{V}_b$  und wir setzen:  $\text{anfang}_0(1) := \text{ende}_0(1) := r$ , wobei  $r$  die Wurzel von  $B$  sei und wir setzen:  $\text{farbe}_0(1) := \text{gelb}$ . Der Baum  $F_0$  ist offensichtlich ein F-Baum zu  $B_0$  über  $\mathcal{V}_b$ .

Für  $i \geq 0$  bestimmen wir  $F_{i+1}$  aus  $F_i$  nach dem folgenden Verfahren:

#### {Kopieren}

- 1:  $F'_i := F_i$
- 2: für jedes  $v \in F_i$
- 3:     wenn  $\text{farbe}_i(v) = \text{grün}$
- 4:          $\text{farbe}'_i := \text{gelb}$

#### {Erweitern}

- 5: für jedes  $v \in F'_i$
- 6:     wähle freien Knotennamen  $w$  und mache  $w$  zum rechtesten Nachfolger von  $v$

- 7:  $\text{farbe}'_i(w) := \text{gelb}$   
 8:  $\text{anfang}'_i(w) := \text{ende}'_i(w) :=$  linker Nachfolger von  $\text{ende}_i(v)$   
 falls kein linker Nachfolger existiert wähle das Symbol  $\perp$   
 9:  $\text{ende}'_i(v) :=$  rechter Nachfolger von  $\text{ende}_i(v)$   
 falls kein rechter Nachfolger existiert wähle das Symbol  $\perp$

**{Ausdünnen}**

- 10: entferne alle Knoten  $v$  aus  $F'_i$  mit  $\text{ende}'_i(v) = \perp$ , die keine Nachfahren  $w$  haben mit  $\text{ende}'_i(w) \neq \perp$

**{Verdichten}**

- 11: solange es noch Knoten in  $F'_i$  gibt mit  $\text{ende}'_i(v) = \perp$   
 12: wähle einen Knoten  $v \in F'_i$  aus mit  $\text{ende}'_i(v) = \perp$   
 13: sei  $w$  der rechteste Nachfahre von  $v$   
 14: mache die Nachfolger von  $w$  zu Nachfolgern von  $v$   
 15:  $\text{ende}'_i(v) := \text{ende}'_i(w)$   
 16:  $\text{farbe}'_i(v) := \text{grün}$   
 17: entferne  $w$  aus  $F'_i$   
 18:  $F_{i+1} := F'_i$

**Erläuterungen** Wird ein Knoten  $w$  aus  $F'_i$  entfernt, so werde  $\text{farbe}'_i(w) := \text{rot}$  gesetzt. Im Erweiterungsschritt wird zu jedem Knoten von  $F_i$  ein neuer Knoten zu  $F'_i$  hinzugefügt. Da der untersuchte Baum  $B$  die Breite  $b$  hat, besitzt  $F_i$  höchstens  $b$  Knoten. Es sind also in  $\mathcal{V}_b$  genügend freie Knotennamen für den Erweiterungsschritt vorhanden. Um eine eindeutige Folge von F-Bäumen zu erhalten, werden bei den Schritten 5 und 12 die Knoten in der Reihenfolge ihrer Knotennamen behandelt. Im Schritt 6 werde der jeweils kleinste freie Knotenname verwendet.

Aus dem folgenden Lemma ergibt sich, daß die nach dem obigen Verfahren bestimmte Folge  $F_0, F_1, \dots$  eine Folge von F-Bäumen zu  $B_0, B_1, \dots$  ist und es dient als Vorbereitung zum Beweis des nachfolgenden Satzes. Der Nachweis der Teilaussagen kann in ähnlicher Form wie für Lemma (4.1) erfolgen.

**Lemma 4.2** Sei  $B$  ein unendlicher  $\{0, 1\}$ -Baum. Sei  $V_i$  die Knotenmenge der  $i$ -ten Ebene von  $B$ . Sei  $F_0, F_1, \dots$  die Folge von Bäumen gemäß obigem Verfahren zu  $B$ . Sei  $i \in \mathbb{N}$ . Sei  $v, w \in F_i$ . Dann gelten die folgenden Aussagen:

- i)  $1 \in F_i$ ,
- ii)  $\text{ende}_i$  ist eine bijektive Abbildung zwischen den Knoten von  $F_i$  und  $V_i$ ,
- iii) ist  $\text{ende}_i(w)$  ein Nachfahre von  $\text{anfang}_i(v)$ , so ist  $w$  ein Nachfahre von  $v$ ,
- iv) ist  $w$  ein Nachfolger von  $v$ , so ist  $\text{anfang}_i(w)$  ein 0-Knoten und ein Nachfahre von  $\text{anfang}_i(v)$ ,

- v) hat  $\text{anfang}_i(v)$  einen Nachfahren in  $V_{i+1}$  und wird  $v$  beim Übergang von  $F_i$  zu  $F_{i+1}$  nicht übernommen, so gilt:  $v \in F_{i+1}$ ,
- vi) ist  $v \in F_{i+1}$ , so gilt:  $\text{ende}_{i+1}(v) \leq^{01} \text{ende}_i(v)1$ ,
- vii)  $v <_{\text{pre}} w \Leftrightarrow \text{ende}(v) <^{01} \text{ende}(w)$ ,
- viii)  $\text{anfang}_i(v) = \text{ende}_i(v)$  oder  $\text{ende}_i(v)$  ist ein Nachfahre von  $\text{anfang}_i(v)$ .

Der folgende Satz gibt uns die Möglichkeit, das beschriebene Verfahren durch einen deterministischen Rabin-Automaten zu simulieren.

**Satz 4.2** Sei  $B$  ein unendlicher  $\{0, 1\}$ -Baum mit beschränkter Breite  $b$ . Sei  $\mathcal{V}_b := \{1, \dots, 2b\}$ . Sei  $F_0, F_1, \dots$  die Folge der F-Bäume gemäß obigem Verfahren zu  $B$ . Dann sind die folgenden Aussagen äquivalent:

- i) Es gibt einen Pfad  $\pi$  in  $B$ , der unendlich oft nach links verzweigt.
- ii) Es gibt einen Knotennamen  $v \in \mathcal{V}_b$  mit  $\text{farbe}_i(v) = \text{grün}$  für unendlich viele  $i \in \mathbb{N}$  und  $\text{farbe}_i(v) = \text{rot}$  für nur endlich viele  $i \in \mathbb{N}$ .

**Beweis:** i) $\Rightarrow$ ii): Sei  $\pi$  ein Pfad von  $B$ , der unendlich oft nach links verzweigt. Wir suchen einen Knotennamen  $v_\pi$ , der unendlich oft grün ist und nur endlich oft rot.

Sei  $U$  die Menge der Knotennamen  $v$ , für die es einen Index  $j$  gibt mit  $v \in C_i$  für alle  $i > j$ . D.h.  $U$  ist die Menge der Knotennamen, die nur endlich oft rot sind. Es bleibt zu zeigen, daß es in  $U$  einen Knotennamen gibt, der unendlich oft grün ist.

Es gilt nach Lemma (4.2) i) für alle  $i \in \mathbb{N}$ :  $1 \in C_i$ , also ist  $1 \in U$ .

Sei  $i \in \mathbb{N}$  derart, daß für alle  $v \in U$  und  $j \geq i$  gilt:  $v \in C_j$ .

Es gilt offensichtlich für alle  $j \in \mathbb{N}$ :  $\text{anfang}_j(1)$  liegt auf  $\pi$ . Sei  $v_\pi$  ein Knoten aus  $U$  mit  $\text{anfang}_i(v_\pi)$  liegt auf  $\pi$  und für alle Nachfahren  $v \in U$  von  $v_\pi$  in  $C_i$  liege  $\text{anfang}_i(v)$  nicht auf  $\pi$ .

Annahme:  $v_\pi$  übernimmt nur endlich oft einen Nachfolger. Dann gibt es ein  $i' > i$  so, daß  $\text{ende}_j(v_\pi)$  ein 1-Knoten ist für alle  $j \geq i'$ . Da  $\pi$  unendlich oft nach linksverzweigt, gibt es ein  $i'' \geq i'$  mit  $\pi(i'')$  ist ein 0-Knoten. In  $C_{i''}$  gibt es nach Lemma (4.2) ii) einen Knoten  $v$  mit  $\text{ende}_{i''}(v) = \pi(i'')$ . Mit Lemma (4.2) iii) folgt:  $v$  ist Nachfahre von  $v_\pi$  in  $C_{i''}$ . Sei  $v'$  der Nachfolger von  $v_\pi$  in  $C_{i''}$  auf dem Pfad von  $v_\pi$  nach  $v$ .

Nach Lemma (4.2) iv) liegt  $\text{anfang}_{i''}(v')$  auf  $\pi$ . Da  $v'$  nicht übernommen wird, gilt nach Lemma (4.2) v):  $v' \in C_j$  für alle  $j \geq i''$ , d.h.  $v' \in U$  im Widerspruch zur Wahl von  $v_\pi$ .

ii) $\Rightarrow$ i): Sei  $v$  ein Knotenname, der unendlich oft grün und nur endlich oft rot ist. Wir suchen einen Pfad  $\pi$  von  $B$ , der unendlich oft nach links verzweigt. Dazu definieren wir eine Knotenfolge  $x_0, x_1, \dots$  von  $B$  derart, daß  $x_{i+1}$  Nachfahre von  $x_i$  ist und  $[x_i, x_{i+1}]$  eine Linksabzweigung enthält für alle  $i \in \mathbb{N}$ . Der durch die  $x_i$  führende Pfad von  $B$  ist dann der gesuchte Pfad  $\pi$ .

Da  $v$  nur endlich oft rot ist, gibt es einen Index  $i_0$  so, daß  $v \in C_j$  gilt für alle  $j \geq i_0$ . Setze  $x_0 := \text{anfang}_{i_0}(v)$ .

Sei  $i_1 > i_0$  mit  $\text{farbe}_{i_1}(v) = \text{grün}$ . Beim Übergang zu  $C_{i_1}$  hat  $v$  einen Nachfolger  $w$  übernommen. Es gilt also:  $\text{anfang}_{i_1-1}(w)$  liegt auf  $\text{pfad}_{i_1}(v)$ , also enthält  $\text{pfad}_{i_1}(v)$  nach Lemma (4.2) iv) einen 0-Knoten.

Da  $B$  eine beschränkte Breite hat, gibt es in der Ebene des Knotens  $\text{anfang}_{i_1-1}(w)$  nur endlich viele Knoten. Diese seien in der Reihenfolge von links nach rechts:  $k_1, k_2, \dots, k_n$ . Für ein  $j \in \{1, \dots, n\}$  gilt also  $k_j = \text{anfang}_{i_1-1}(w)$ . Für alle  $l \geq i_1$  liegt einer der Knoten  $k_1, \dots, k_n$  auf  $\text{pfad}_l(v)$ , bezeichne diesen Knoten mit  $k(l)$ . Mit Lemma (4.2) vi) folgt:  $k(l+1) \leq^{01} k(l)$  für alle  $l \geq i_1$ . Es gibt also  $j_1, l_1 \in \mathbb{N}$  derart, daß gilt:

(\*) Für alle  $i > l_1$  liegt  $k_{j_1}$  auf  $\text{pfad}_i(v)$ .

Setze  $x_1 := k_{j_1}$ . Sei nun  $i_2$  ein Index mit  $i_2 > l_1$  und  $\text{farbe}_{i_2}(v) = \text{grün}$ . Beim Übergang zu  $C_{i_2}$  hat  $v$  einen Nachfolger  $w$  übernommen, d.h.  $\text{pfad}_{i_2}(v)$  enthält den 0-Knoten  $\text{anfang}_{i_2-1}(w)$ . Wegen (\*) ist  $\text{tiefe}(\text{anfang}_{i_2-1}(w)) > \text{tiefe}(x_1)$ . Es gibt (mit derselben Argumentation wie für den Knoten  $x_1$ ) einen Knoten  $x_2 \leq \text{anfang}_{i_2-1}(w)$  in der Ebene von  $\text{anfang}_{i_2-1}(w)$  und einen Index  $l_2 > l_1$  mit:

Für alle  $i > l_2$  liegt  $x_2$  auf  $\text{pfad}_i(v)$ .

Fahren wir auf diese Weise fort, erhalten wir eine Knotenfolge  $x_0, x_1, \dots$  von  $B$  so, daß für alle  $i \in \mathbb{N}$  gilt:

- $x_{i+1}$  ist Nachfahre von  $x_i$  und
- $[x_i, x_{i+1}]$  enthält eine Linksabzweigung.

Der durch die  $x_i$  führende Pfad  $\pi$  zweigt also unendlich oft nach links ab.  $\square$

Die Beschriftungskomponente anfang eines F-Baums dient nur der Vereinfachung der Argumentation im Beweis von Satz (4.2). Wir werden diese Komponente bei der späteren Anwendung des Verfahrens weglassen.

## 5 Determinisierungsverfahren

Wir stellen in diesem Abschnitt drei Verfahren vor, die aus einem gegebenen nichtdeterministischen Büchi-Automaten einen deterministischen Rabin-Automaten konstruieren. Dabei fassen wir ein Determinisierungsverfahren als die Anwendung eines Baumuntersuchungsverfahrens auf einen der in Abschnitt (3) eingeführten reduzierten Berechnungsbäume auf.

## 5.1 Verfahren nach Muller und Schupp

Das Determinisierungsverfahren nach Muller und Schupp wendet die Strategie der Pfadsegmentierung (siehe Abschnitt (4.2)) auf reduzierte 01-Berechnungsbäume, wie sie in Abschnitt (3.1) eingeführt wurden, an. D.h. die Zustände des konstruierten deterministischen Automaten  $\mathcal{D}_M(\mathcal{A}) = (Q', \Sigma, q'_I, \delta', \Omega)$  zu einem gegebenen nichtdeterministischen Büchi-Automaten  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  sind im wesentlichen S-Bäume. Der Lauf von  $\mathcal{D}_M(\mathcal{A})$  auf einem Wort  $\alpha \in \Sigma^\omega$  entspricht der Folge der S-Bäume  $S_0, S_1, \dots$  zu  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$ . Die Akzeptierbedingung  $\Omega$  von  $\mathcal{D}_M(\mathcal{A})$  ergibt sich aus Satz (4.1):  $\Omega$  ist eine Rabin-Bedingung, die zu jedem vorkommenden Knotennamen  $v$  ein Rabin-Paar  $(G_v, R_v)$  beinhaltet. Dabei besteht die Menge  $G_v$  aus denjenigen S-Bäumen, in denen  $v$  grün und  $R_v$  besteht aus denjenigen S-Bäumen, in denen  $v$  rot ist.

In einem reduzierten 01-Berechnungsbaum  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$  gibt es zwischen den Knoten der Ebene  $i$  und den Beschriftungen dieser Knoten eine bijektive Beziehung, wir identifizieren also die Beschriftung mit dem beschrifteten Knoten. Im verwendeten S-Baum zur Ebene  $i$  von  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$  ordnen wir einem Blatt also keinen Knoten dieser Ebene zu, sondern dessen Beschriftung, d.h. eine Teilmenge der Zustandsmenge  $Q$  von  $\mathcal{A}$ .

Beim Übergang von  $S_i$  zu  $S_{i+1}$  bestimmen wir im Erweiterungsschritt zu den Blättern von  $S_i$  die Nachfolger. D.h. wir müssen zu der Beschriftung eines Knotens  $v$  der Ebene  $i$  von  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$  die Beschriftung der Nachfolger von  $v$  berechnen. Bei der Bestimmung von  $V_{i+1}$  aus  $V_i$  in der Definition von  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$  benötigen wir eine Ordnung unter den Knoten von  $V_i$ , um zu einem auftretenden Zustand  $q$  das linkeste Vorkommen ermitteln zu können. Die in Abschnitt (4.2) eingeführten S-Bäume stattdessen wir deshalb mit einer Geschwisterordnung aus, die uns ermöglicht die Ordnung zwischen den Knoten in  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$  nachzubilden.

Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein nichtdeterministischer Büchi-Automat. Es sei die Menge  $\mathcal{V} = \{1, \dots, 3|Q| - 1\}$  von Knotennamen gegeben. Die Zustände des deterministischen Rabin-Automaten  $\mathcal{D}_M(\mathcal{A}) = (Q', \Sigma, q'_I, \delta', \Omega)$  zu  $\mathcal{A}$  nach Muller und Schupp sind geordnete beschriftete Bäume über  $\mathcal{V}$ .

Die Beschriftung eines Baums  $C \in Q'$  hat zwei Komponenten: *zust* und *farbe*. Die Komponente *zust* ordnet jedem Blatt von  $C$  eine nichtleere Teilmenge von  $Q$  zu. Die Komponente *farbe* ordnet jedem Knotennamen aus  $\mathcal{V}$  eine Farbe aus  $\{rot, gelb, grün\}$  zu. Weiterhin gelten die folgenden Bedingungen:

1. ein Knoten  $v$  von  $C$  hat genau zwei Nachfolger oder  $v$  ist ein Blatt,
2. für zwei verschiedene Blätter  $v, w$  von  $C$  gilt:  $zust(v) \cap zust(w) = \emptyset$  und
3. für ein  $v \in \mathcal{V}$  mit  $v \notin C$  gilt:  $farbe(v) = rot$ .

Die folgende Bemerkung gibt uns eine Abschätzung der Größe des Automaten  $\mathcal{D}_M(\mathcal{A})$ .

**Bemerkung 5.1** Es gilt  $|Q'| \in 2^{\mathcal{O}(|Q| \log |Q|)}$ .

**Beweis:** Die Beschriftungskomponente  $\text{zust}$  eines Baumes  $C \in Q'$  lässt sich als Funktion von  $\lambda : Q \rightarrow \mathcal{V} \cup \{\perp\}$  darstellen. Gibt es zu einem  $q \in Q$  ein Blatt  $v$  von  $C$  mit  $q \in \text{zust}(v)$ , so ist  $v$  wegen Bedingung 2. eindeutig und wir setzen  $\lambda(q) := v$ . Gibt es kein solches Blatt  $v$ , so setzen wir  $\lambda(q) := \perp$ . Es gibt  $(3|Q|)^{|Q|} \in 2^{O(|Q|\log|Q|)}$  solche Funktionen.

Die Beschriftungskomponente  $\text{farbe}$  stellen wir durch eine Funktion von  $\mathcal{V}$  nach  $\{\text{rot}, \text{grün}, \text{gelb}\}$  dar. Wir erhalten eine mögliche Anzahl von  $3^{(3|Q|-1)} \in 2^{O(|Q|\log|Q|)}$  solcher Funktionen.

Nach Bemerkung (4.1) gibt es  $2^{O(|Q|\log|Q|)}$  verschiedene geordnete Bäume über  $\mathcal{V}$ . Insgesamt gilt also  $|Q'| \in 2^{O(|Q|\log|Q|)}$ .  $\square$

Der Anfangszustand  $q_I'$  von  $\mathcal{D}$  besteht nur aus dem Knoten  $1 \in \mathcal{V}$  mit  $\text{zust}(1) = \{q_I\}$  und  $\text{farbe}(1) = \text{grün}$  falls  $q_I \in F$  und  $\text{farbe}(1) = \text{rot}$  sonst.

Sind ein  $C \in Q'$  und ein  $a \in \Sigma$  gegeben, so erhalten wir  $\delta'(C, a)$  durch Anwendung des folgenden Verfahrens aus  $C$ :

### {Kopieren}

- 1:  $C' := C$
- 2: für jedes  $v \in C$
- 3:     wenn  $\text{farbe}(v) = \text{grün}$
- 4:          $\text{farbe}'(v) := \text{gelb}$

### {Erweitern}

- 5: für jedes Blatt  $v \in C$
- 6:      $\text{zust}'(v) := \delta(\text{zust}(v), a)$
- 7:  $N := \{q \mid q \in \text{zust}'(v) \text{ für ein Blatt } v \in C'\}$
- 8: gilt für ein  $q \in N$ :  $q \in \text{zust}'(v)$  und  $q \in \text{zust}'(w)$   
mit  $v$  ist links von  $w$ , so entferne  $q$  aus  $\text{zust}'(w)$
- 9: für jedes Blatt  $v \in C$
- 10:    wenn  $\text{zust}'(v) \cap F \neq \emptyset$
- 11:       füge neuen Nachfolger  $w_0$  von  $v$  zu  $C'$  hinzu
- 12:        $\text{zust}'(w_0) := \text{zust}'(v) \cap F$ ;  $\text{farbe}'(w_0) := \text{grün}$
- 13:    wenn  $\text{zust}'(v) \cap F^c \neq \emptyset$
- 14:       füge neuen Nachfolger  $w_1$  von  $v$  zu  $C'$  hinzu
- 15:        $\text{zust}'(w_1) := \text{zust}'(v) \cap F^c$ ;  $\text{farbe}'(w_1) := \text{rot}$
- 16:    wenn  $v$  zwei neue Nachfolger bekommen hat, sei  $w_0 < w_1$  in  $C'$
- 17:     $\text{zust}'(v) := \emptyset$

### {Ausdünnen}

- 18: entferne alle Knoten  $v \in C$  aus  $C'$ , die kein neues Blatt als Nachfahren haben

### {Verdichten}

- 19: solange es noch Knoten in  $C'$  gibt, die genau einen Nachfolger haben
- 20:    wähle einen Knoten  $v \in C'$  aus, der genau einen Nachfolger  $w$  hat
- 21:    wenn  $\text{farbe}'(w) \in \{\text{grün}, \text{gelb}\}$

- 22:  $\text{farbe}'(v) := \text{grün}$   
 23: mache Nachfolger von  $w$  zu Nachfolgern von  $v$  und  
 übernehme deren Ordnung untereinander  
 24:  $\text{zust}'(v) := \text{zust}'(w)$   
 25: entferne  $w$  aus  $C'$   
 26:  $\delta'(C, a) := C'$

**Erläuterungen** Wird ein Knoten  $v$  aus  $C'$  entfernt, so werde  $\text{farbe}'(v) := \text{rot}$  gesetzt. Da für ein Blatt  $v$  von  $C$  gilt:  $\text{zust}(v) \neq \emptyset$  und für verschiedene Blätter  $v, w$  von  $C$  gilt:  $\text{zust}(v) \cap \text{zust}(w) = \emptyset$ , besitzt  $C$  höchstens  $|Q|$  Blätter. Wie im zugrundeliegenden Baumuntersuchungsverfahren ergibt sich, daß für den Erweiterungsschritt genügend freie Knotennamen in  $\mathcal{V}$  vorhanden sind. Um  $\delta'$  deterministisch zu machen, wählen wir in den Schritten 9 und 20 unter den möglichen Knoten den Knoten mit dem kleinsten Knotennamen aus. In den Schritten 11 und 14 wählen wir den kleinsten freien Knotennamen.

Wird im Erweiterungsschritt ein neues Blatt  $v$  erzeugt, so gilt:  $\text{zust}'(v) \neq \emptyset$ . Ein Knoten  $v$  aus  $C$ , der kein neues Blatt als Nachfolger hat, wird im Schritt 18 aus  $C'$  entfernt. Nach Schritt 18 gilt also für jedes Blatt  $v$  von  $C'$ :  $\text{zust}'(v) \neq \emptyset$ . Durch Schritt 24 bleibt diese Bedingung für Blätter von  $C'$  erhalten.

Nach den Schritten 19 – 24 gibt es  $C'$  keine Knoten, die genau einen Nachfolger haben. Da im Erweiterungsschritt jedes Blatt höchstens zwei neue Nachfolger erhält, kann kein Knoten mit mehr als zwei Nachfolgern entstehen. Ein Knoten  $v$  von  $\delta'(C, a)$  hat also genau zwei Nachfolger oder  $v$  ist ein Blatt.

Nach Schritt 8 gilt für jedes  $q \in N$  für genau ein Blatt  $v \in C'$ :  $q \in \text{zust}'(v)$ . Diese Bedingung wird durch die folgenden Schritte nicht berührt. Es gilt also für zwei verschiedene Blätter  $v$  und  $w$  von  $\delta'(C, a)$ :  $\text{zust}(v) \cap \text{zust}(w) = \emptyset$ .

Wird ein Knoten  $v$  aus  $C'$  entfernt, so wird  $\text{farbe}'(v) := \text{rot}$  gesetzt. Damit ist  $\delta'(C, a)$  ein Element von  $Q'$ .

Die Akzeptierbedingung  $\Omega$  enthält zu jedem Knotennamen  $v \in \mathcal{V}$  ein Rabin-Paar  $(G_v, R_v)$ . Die Menge  $G_v$  enthält alle Bäume  $C \in Q'$ , in denen  $v$  grün ist und  $R_v$  enthält alle Bäume, in denen  $v$  rot ist.

Das folgende Lemma stellt die vorher beschriebene Beziehung zwischen dem Baumuntersuchungsverfahren durch Pfadsegmentierungsbäume und dem konstruierten deterministischen Automaten her. Der Beweis ergibt sich direkt durch einen Vergleich des Erweiterungsschritts (insbesondere Schritte 5-8) mit dem Übergang von einer Ebene zur nächsten im reduzierten 01-Berechnungsbaum (siehe Abschnitt (3)).

**Lemma 5.1** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein Büchi-Automat. Sei  $b = 3|Q| - 1$  und  $\mathcal{V} = \{1, \dots, b\}$ . Sei  $\mathcal{D} = (Q', \Sigma, q'_I, \delta', \Omega)$  der nach dem oben beschriebenen Verfahren aus  $\mathcal{A}$  konstruierte Rabin-Automat. Sei  $\alpha \in \Sigma^\omega$ . Sei  $S_0, S_1, \dots$  die Folge der Segmentierungsbäume zu  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$ . Sei  $C_0, C_1, \dots$  der Lauf von  $D$  auf  $\alpha$ . Dann gilt für alle  $i \in \mathbb{N}$  und  $v, w \in \mathcal{V}$ :

$$\text{i) } (V_{C_i}, E_{C_i}) = (V_{S_i}, E_{S_i}),$$

- ii)  $\text{farbe}_{C_i}(v) = \text{farbe}_{S_i}(v)$ ,
- iii) sind  $v$  und  $w$  Blätter von  $S_i$ , so gilt:  $\text{knoten}_{S_i}(v) < \text{knoten}_{S_i}(w) \Leftrightarrow v < w$ ,
- iv) ist  $v$  ein Blatt von  $S_i$ , so gilt:  $\text{knoten}_{C_i}(v) = \tilde{B}_{\mathcal{A},\alpha}^{01}(\text{knoten}_{S_i}(v))$ .

Mit Lemma (5.1) ii) und Satz (4.1) erhalten wir

**Satz 5.1** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein nichtdeterministischer Büchi-Automat. Dann gilt  $L(\mathcal{A}) = L(\mathcal{D}_M(\mathcal{A}))$ .

## 5.2 Verfahren nach Emerson und Sistla

Das Determinisierungsverfahren nach Emerson und Sistla wendet die Strategie der Pfadverfolgung auf Standardberechnungsbäume an. Die in [ES84] betrachteten nichtdeterministischen Büchi-Automaten  $\mathcal{A}$  hatten die Eigenschaft, daß ein Zustand  $q$  von  $\mathcal{A}$  mit einem Wort  $\beta \in \Sigma^*$  nur auf höchstens eine Weise erreicht werden konnte, daß also für alle  $\alpha \in \Sigma^\omega$ ,  $i \in \mathbb{N}$  und  $q \in Q$  gilt: (\*)  $|R_{i,q}^\alpha| \leq 1$ . Das bedeutet insbesondere, daß der Standardberechnungsbaum  $B_{\mathcal{A},\alpha}$  höchstens die Breite  $|Q|$  besitzt. Mit den in Abschnitt (3) beschriebenen Reduktionen, kann auf die Bedingung (\*) verzichtet werden.

Aus den betrachteten Büchi-Automaten  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$ , die die Bedingung (\*) erfüllen, entstehen nach dem Verfahren von Emerson und Sistla deterministische Rabin-Automaten mit  $2^{\mathcal{O}(|Q|^2)}$  Zuständen und  $|Q|^2$  Rabin-Paaren, siehe [ES84].

## 5.3 Verfahren durch vereinfachte Pfadverfolgung

Das hier beschriebene Determinisierungsverfahren wendet die Strategie der Pfadverfolgung für  $\{0, 1\}$ -Bäume aus Abschnitt (4.4) auf reduzierte 01-Berechnungsbäume, wie sie in Abschnitt (3.1) dargestellt wurde, an. Die Zustände des konstruierten deterministischen Automaten  $\mathcal{D}_F(\mathcal{A}) = (Q', \Sigma, q'_I, \delta', \Omega)$  zu einem gegebenen nichtdeterministischen Büchi-Automaten  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  sind F-Bäume. Der Lauf von  $\mathcal{D}_F(\mathcal{A})$  auf einem Wort  $\alpha \in \Sigma^\omega$  entspricht der Folge der F-Bäume  $F_0, F_1, \dots$  zu  $\tilde{B}_{\mathcal{A},\alpha}^{01}$ . Die Akzeptierbedingung  $\Omega$  von  $\mathcal{D}_F(\mathcal{A})$  ist eine Rabin-Bedingung, die zu jedem der verwendeten Knotennamen  $v$  ein Rabin-Paar  $(G_v, R_v)$  enthält. Die Menge  $G_v$  besteht aus denjenigen Bäumen aus  $Q'$ , in denen  $v$  grün ist und  $R_v$  aus denjenigen, in denen  $v$  rot ist.

In einem reduzierten 01-Berechnungsbaum  $\tilde{B}_{\mathcal{A},\alpha}^{01}$  gibt es zwischen den Knoten der Ebene  $i$  und den Beschriftungen dieser Knoten eine bijektive Beziehung, wir identifizieren also die Beschriftung mit dem beschrifteten Knoten. Im verwendeten F-Baum  $F$  zur Ebene  $i$  von  $\tilde{B}_{\mathcal{A},\alpha}^{01}$  ordnen wir einem Knoten  $v \in F$  die Teilmenge von  $Q$  zu, mit der der zu  $v$  gehörende Knoten der Ebene  $i$  von  $\tilde{B}_{\mathcal{A},\alpha}^{01}$  beschriftet ist.

Die in Kapitel (4.4) eingeführten F-Bäume sind mit einer Geschwisterordnung ausgestattet. Ist ein endlicher  $\{0, 1\}$ -Baum  $B$  der Tiefe  $t$  und ein dazugehöriger F-Baum  $F$  gegeben, so repräsentiert jeder Knoten  $v$  von  $F$  ein Blatt der Tiefe  $t$  von  $B$  nämlich

ende( $v$ ). Sei  $<_{\text{post}}$  die gemäß postorder Durchlauf auf ganz  $F$  erweiterte Geschwisterordnung und seien  $v_1 <_{\text{post}} v_2 <_{\text{post}} \dots <_{\text{post}} v_l$  die Knoten von  $F$ . Dann gilt  $\text{ende}(v_1) <^{01} \text{ende}(v_2) <^{01} \dots <^{01} \text{ende}(v_l)$ , siehe Lemma (4.2). Ist ein F-Baum  $F_i$  zur  $i$ -ten Ebene eines reduzierten 01-Berechnungsbaums  $\tilde{B}_{\mathcal{A},\alpha}^{01}$  gegeben, so können wir also mit Hilfe der Geschwisterordnung von  $F_i$  das Bestimmen des linkesten Vorkommens eines Zustands  $q \in Q$  in der Ebene  $i + 1$  von  $\tilde{B}_{\mathcal{A},\alpha}^{01}$  nachbilden.

Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein nichtdeterministischer Büchi-Automat. Sei die Menge  $\mathcal{V} = \{1, \dots, 2|Q|\}$  von Knotennamen gegeben. Die Zustände des deterministischen Rabin-Automaten  $\mathcal{D}_F(\mathcal{A}) = (Q', \Sigma, q'_I, \delta', \Omega)$  sind geordnete beschriftete Bäume über  $\mathcal{V}$ .

Die Beschriftung eines Baums  $C \in Q'$  hat die Komponenten: *zust* und *farbe*. Die Komponente *zust* ordnet jedem Knoten von  $C$  einen nichtleere Teilmenge von  $Q$  zu. Die Komponente *farbe* ordnet jedem Knotennamen aus  $\mathcal{V}$  eine Farbe aus  $\{\text{rot}, \text{gelb}, \text{grün}\}$  zu. Weiterhin gelten die folgenden Bedingungen:

1. für zwei verschiedene Knoten  $v, w$  von  $C$  gilt:  $\text{zust}(v) \cap \text{zust}(w) = \emptyset$  und
2. für ein  $v \in \mathcal{V}$  gilt:  $v \notin C \Leftrightarrow \text{farbe}(v) = \text{rot}$ .

Wir erhalten die folgende Bemerkung über die Mächtigkeit von  $Q'$ .

**Bemerkung 5.2** Es gilt  $|Q'| \in 2^{\mathcal{O}(|Q| \log |Q|)}$ .

**Beweis** Wie bei Bemerkung (5.1)  $\square$

Der Anfangszustand  $q'_I$  von  $\mathcal{D}_F(\mathcal{A})$  besteht nur aus dem Knoten  $1 \in \mathcal{V}$  mit  $\text{zust}(1) = \{q_I\}$  und  $\text{farbe}(1) = \text{gelb}$ . Offensichtlich gilt  $q'_I \in Q'$ .

Sind ein  $C \in Q'$  und ein  $a \in \Sigma$  gegeben, so erhalten wir  $\delta'(C, a)$  durch Anwendung des folgenden Verfahrens aus  $C$ :

### {Kopieren}

- 1:  $C' := C$
- 2: für jedes  $v \in C$
- 3:     wenn  $\text{farbe}'(v) = \text{grün}$
- 4:          $\text{farbe}'(v) := \text{gelb}$

### {Erweitern}

- 5: für jeden Knoten  $v \in C$
- 6:      $\text{zust}'(v) := \delta(\text{zust}(v), a)$
- 7:  $N := \{q \mid q \in \text{zust}'(v) \text{ für ein } v \in C'\}$
- 8: gilt für ein  $q \in N$ :  $q \in \text{zust}'(v)$  und  $q \in \text{zust}'(w)$   
mit  $v <_{\text{post}} w$ , so entferne  $q$  aus  $\text{zust}'(w)$
- 9: für jedes  $v \in C'$
- 10:     wähle freien Knotennamen  $w$  und mache  $w$  zum rechtesten Nachfolger von  $v$
- 11:      $\text{farbe}'(w) := \text{gelb}$
- 12:      $\text{zust}'(w) := \text{zust}'(v) \cap F$

13:  $\text{zust}'(v) := \text{zust}'(v) \cap F^c$

**{AUSDÜNNEN}**

14: entferne alle Knoten  $v$  aus  $C'$  mit  $\text{zust}'(v) = \emptyset$ , die keine Nachfahren  $w$  haben mit  $\text{zust}'(w) \neq \emptyset$

**{VERDICHTEN}**

- 15: solange es noch Knoten in  $C'$  gibt, mit  $\text{zust}'(v) = \emptyset$   
 16: wähle einen Knoten  $v \in C'$  aus mit  $\text{zust}'(v) = \emptyset$   
 17: sei  $w$  der rechteste Nachfahre von  $v$   
 18: mache die Nachfolger von  $w$  zu Nachfolgern von  $v$   
 19:  $\text{zust}'(v) := \text{zust}'(w)$   
 20:  $\text{farbe}'(v) := \text{grün}$   
 21: entferne  $w$  aus  $C'$   
 22:  $\delta'(C, a) := C'$

**Erläuterungen** Wird ein Knoten  $w$  aus  $C'$  entfernt, so werde  $\text{farbe}'(w) := \text{rot}$  gesetzt. Um  $\delta'$  deterministisch zu machen, wählen wir in den Schritten 5 und 16 unter den möglichen Knoten den Knoten mit dem kleinsten Knotennamen aus. Im Schritt 10 wählen wir den kleinsten freien Knotennamen aus.

Nach dem Ausdünnen und Verdichten gibt es keine Knoten  $v \in C'$  mit  $\text{zust}'(v) = \emptyset$ . Nach dem Erweiterungsschritt gilt offensichtlich für zwei verschiedene Knoten  $v, w \in C'$ :  $\text{zust}'(v) \cap \text{zust}'(w) = \emptyset$ . Kommt ein neuer Knoten  $v$  zu  $C'$  hinzu, so wird  $\text{farbe}'(v) := \text{gelb}$  gesetzt, wird ein Knoten  $w$  aus  $C'$  entfernt, so wird  $\text{farbe}'(w) := \text{rot}$  gesetzt. Damit ist  $\delta'(C, a)$  wiederum ein Element von  $Q'$ .

Die Akzeptierbedingung  $\Omega$  enthält zu jedem Knotennamen  $v \in \mathcal{V}$  ein Rabin-Paar  $(G_v, R_v)$ . Die Menge  $G_v$  besteht aus denjenigen Bäumen  $C \in Q'$ , in denen  $v$  grün ist und  $R_v$  aus denjenigen Bäumen in denen  $v$  rot ist.

Mit Hilfe des folgenden Lemmas stellen wir die Beziehung zwischen der Strategie der Pfadverfolgung von  $\{0, 1\}$ -Bäumen und dem konstruierten Rabin-Automaten her. Der Beweis ergibt sich durch einen Vergleich des Erweiterungsschritts (insbesondere Schritte 5-8) mit dem Übergang von einer Ebene zur nächsten im reduzierten 01-Berechnungsbaum (siehe Abschnitt (3)).

**Lemma 5.2** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein Büchi-Automat. Sei  $\mathcal{V} = \{1, \dots, 2|Q|\}$ . Sei  $\alpha \in \Sigma^\omega$ . Sei  $F_0, F_1, \dots$  die Folge der Pfadverfolgungsbäume zu  $\tilde{B}_{\mathcal{A}, \alpha}^{01}$ . Sei  $C_0, C_1, \dots$  der Lauf von  $\mathcal{D}_F(\mathcal{A})$  auf  $\alpha$ . Dann gilt für alle  $i \in \mathbb{N}$  und  $v, w \in \mathcal{V}$ :

- i)  $(V_{C_i}, E_{C_i}) = (V_{F_i}, E_{F_i})$ ,
- ii)  $\text{farbe}_{C_i}(v) = \text{farbe}_{F_i}(v)$ ,
- iii) sind  $v$  und  $w$  Knoten von  $F_i$ , so gilt:  $\text{ende}_{F_i}(v) <_{\text{post}} \text{ende}_{F_i}(w) \Leftrightarrow v < w$ ,
- iv) ist  $v$  ein Knoten von  $S_i$ , so gilt:  $\text{zust}_{C_i}(v) = \tilde{B}_{\mathcal{A}, \alpha}^{01}(\text{ende}_{S_i}(v))$ .

Mit Lemma (5.2) ii) und Satz (4.2) erhalten wir

**Satz 5.2** Sei  $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$  ein nichtdeterministischer Büchi-Automat. Dann gilt  $L(\mathcal{A}) = L(\mathcal{D}_F(\mathcal{A}))$ .

## Fazit

In dieser Arbeit wurde das Akzeptieren von Büchi-Automaten durch das Vorhandensein von Pfaden bestimmten Typs in Berechnungsbäumen charakterisiert. Auf der Grundlage von Baumuntersuchungsverfahren wurden zwei optimale Verfahren zur Determinisierung von Büchi-Automaten dargestellt. Die dargestellten Verfahren ermöglichen ein intuitives Verständnis für die Arbeitsweise der konstruierten deterministischen Automaten. Dieses direkte Verständnis war bei dem bisherigen Standardverfahren für die Determinisierung von Büchi-Automaten nach Safra [Saf88], das ebenfalls eine optimale Komplexität der konstruierten Automaten erreichte, nicht gegeben.

Ich bin der Meinung, daß sich das Determinisierungsverfahren, das sich aus der Arbeit von Muller und Schupp [MS95] ableitet, besser zur Darstellung eines optimalen Verfahrens eignet als die übrigen erwähnten Verfahren.

Während der Untersuchung von Determinisierungsverfahren für Büchi-Automaten zu dieser Arbeit, stellte sich heraus, daß auch das Verfahren von Safra als Baumuntersuchungsverfahren interpretiert werden kann. Jedoch war es nicht gelungen, den tatsächlich untersuchten Baum, der sich bei den zuvor beschriebenen Verfahren auf natürliche Weise ergab, ohne Verlust der Anschauung darzustellen. Dies führte dazu, daß diese Sichtweise des Verfahrens von Safra nicht in die Darstellung aufgenommen wurde.

Safra stellte in [Saf92] ein optimales Determinisierungsverfahren für  $\omega$ -Automaten mit Streett-Akzeptierbedingung [Str82] dar. Dieses Verfahren entzieht sich noch mehr einem direkten Verständnis als das Determinisierungsverfahren von Safra für Büchi-Automaten. Die Arbeit von Muller und Schupp [MS95] ergibt auch für Streett-Automaten ein Determinisierungsverfahren. Ob auch hier Vorteile in Bezug auf die intuitive Erfäßbarkeit gegenüber dem Verfahren von Safra existieren, ist eine sich an diese Arbeit anschließende Fragestellung.

## Literatur

- [Büc62] J. Richard Büchi. On a decision method in restricted second order arithmetic. In *International Congress on Logic, Methodology and Philosophy of Science*, pages 1–11. Stanford University Press, 1962.
- [ES84] E. Allen Emerson and A. Prasad Sistla. Deciding full branching time logic. *Information and Control*, 61(3):175–201, 1984.
- [HU79] John E. Hopcroft and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison Wesley, 1979.
- [Löd99] Christof Löding. Optimal bounds for the transformation of omega-automata. In C. Pandu Rangan, V. Raman, and R. Ramanujam, editors, *Foundations of Software Technology and Theoretical Computer Science*, volume 1738 of *Lecture Notes in Computer Science*, pages 97–109. Springer-Verlag, December 1999.
- [McN66] Robert McNaughton. Testing and generating infinite sequences by a finite automaton. *Information and Control*, 9(5):521–530, 1966.
- [Mic88] Max Michel. Complementation is more difficult with automata on infinite words. CNET, Paris, 1988.
- [MS95] David E. Muller and Paul E. Schupp. Simulating alternating tree automata by nondeterministic automata: New results and new proofs of the theorems of Rabin, McNaughton and Safra. *Theoretical Computer Science*, 141(1&2):69–107, 1995.
- [Mul63] David E. Muller. Infinite sequences and finite machines. In *Proceedings of the 4th IEEE Symposium on Switching Circuit Theory and Logical Design*, pages 3–16, 1963.
- [Rab69] Michael O. Rabin. Decidability of second-order theories and automata on infinite trees. *Transactions of the American Mathematical Society*, 141:1–35, July 1969.
- [RS59] Michael O. Rabin and D. Scott. Finite automata and their decision problems. *IBM Journal of Research and Development*, 3:114–125, April 1959.
- [Saf88] Shmuel Safra. On the complexity of omega-automata. In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science, FoCS '88*, pages 319–327, Los Alamitos, California, October 1988. IEEE Computer Society Press.

- [Saf92] Shmuel Safra. Exponential determinization for omega-automata with strong-fairness acceptance condition (extended abstract). In *Proceedings of the 24th Annual ACM Symposium on the Theory of Computing, STOC '92*, pages 275–282, Victoria, British Columbia, Canada, 1992.
- [Str82] Robert S. Streett. Propositional dynamic logic of looping and converse is elementary decidable. *Information and Control*, 54:121–141, 1982.
- [Tho97] Wolfgang Thomas. *Handbook of Formal Language Theory*, volume III, chapter Languages, Automata, and Logic, pages 389–455. Springer-Verlag, 1997.

## **Erklärung**

Hiermit erkläre ich, daß ich diese Arbeit selbständig verfaßt habe. Ich habe dabei nur die angegebenen Hilfsmittel und Quellen verwendet.

Kiel, den 7. Juni 2001

Detlef Kähler